

综合运用python知识

制作FizzBuzz小游戏

试讲人：梁明浩

课堂内容

- 自我介绍
- 游戏介绍
- 技术要点
- 游戏实现
- 课堂总结

自我介绍

姓名：梁明浩

爱好：编程、游戏、电影、读书

热身小游戏

规则：

1. 全班轮流报数
2. 数到7的倍数要改为说“过”

什么是FizzBuzz

- 从1数到100
- 是三的倍数就喊“Fizz! ”
- 是五的倍数就喊“Buzz! ”
- 是十五的倍数就喊“FizzBuzz! ”

知识点

用 `for` 循环输出1到100

```
for i in range(1,101):  
    print(i)
```

```
1  
2  
3  
...  
...  
98  
99  
100
```

知识点

判断n的倍数：取余数运算 %

```
print(1%3)    ### 1
print(2%3)    ### 2
print(3%3)    ### 0
print(4%3)    ### 1
print(5%3)    ### 2
print(6%3)    ### 0
```

```
print(1%3==0) ### False
print(2%3==0) ### False
print(3%3==0) ### True
print(4%3==0) ### False
print(5%3==0) ### False
print(6%3==0) ### True
```

知识点

if-elif-else 条件判断

90 ~ 100分: A
80 ~ 90分: B
70 ~ 80分: C
60 ~ 70分: D
60以下: 不及格

```
if 90 <= 分数 < 100:  
    print("A")  
elif 80 <= 分数:  
    print("B")  
elif 70 <= 分数:  
    print("C")  
elif 60 <= 分数:  
    print("D")  
else:  
    print("不及格")
```


代码实例

```
for i in range(1, 16):  
    if i % 3 == 0:  
        print("Fizz")  
    if i % 5 == 0:  
        print("Buzz")  
    if i % 15 == 0:  
        print("FizzBuzz")  
    else:  
        print(i)
```

发现错误了吗？
该如何修改代码？

```
1  
2  
Fizz  
3  
4  
Buzz  
5  
Fizz  
6  
7  
8  
Fizz  
9  
Buzz  
10  
11  
Fizz  
12  
13  
14  
Fizz  
Buzz
```

代码实例（改正）

```
for i in range(1, 16):  
    if i % 15 == 0:  
        print("FizzBuzz")  
    elif i % 3 == 0:  
        print("Fizz")  
    elif i % 5 == 0:  
        print("Buzz")  
    else:  
        print(i)
```

注意 `if` 判断的优先级

```
1  
2  
Fizz  
4  
Buzz  
Fizz  
7  
8  
Fizz  
Buzz  
11  
Fizz  
13  
14  
FizzBuzz
```

代码实例 (优化)

```
for i in range(1, 16):  
    output = ""  
    if i % 3 == 0:  
        output = "Fizz"  
    if i % 5 == 0:  
        output += "Buzz"  
  
    print(output or i)
```

运用字符串拼接技巧
增加程序的灵活性和扩展性

```
1  
2  
Fizz  
4  
Buzz  
Fizz  
7  
8  
Fizz  
Buzz  
11  
Fizz  
13  
14  
FizzBuzz
```

代码实例 (对比)

```
for i in range(1,16):  
    output = ""  
    if i % 3 == 0:  
        output = "Fizz"  
    if i % 5 == 0:  
        output += "Buzz"  
  
    # 可以增加更多的判断条件  
    # 而不破坏原本的程序结构  
  
    print(output or i)
```

```
for i in range(1,16):  
    if i % 15 == 0:  
        print("FizzBuzz")  
    elif i % 3 == 0:  
        print("Fizz")  
    elif i % 5 == 0:  
        print("Buzz")  
    else:  
        print(i)
```

代码实例 (进一步优化?)

```
for i in range(1,16):  
    output = "fizz" if (i % 3 == 0) else ""  
    output += "buzz" if (i % 5 == 0) else ""  
  
    print(output or i)
```

运用三元操作符精简代码

程序运行逻辑更清晰

```
for i in range(1,16):  
    print( ("fizz" * (i % 3 == 0) + "buzz"*(i % 5 == 0)) or i)
```

运用字符串操作技巧，进一步精简代码

但是逻辑不直观

课堂总结

- `for` 循环
- `%` 运算符取余数
- `if-elif-else` 条件判断
- 条件判断的优先级
- 程序的可扩展性和可读性

谢谢大家