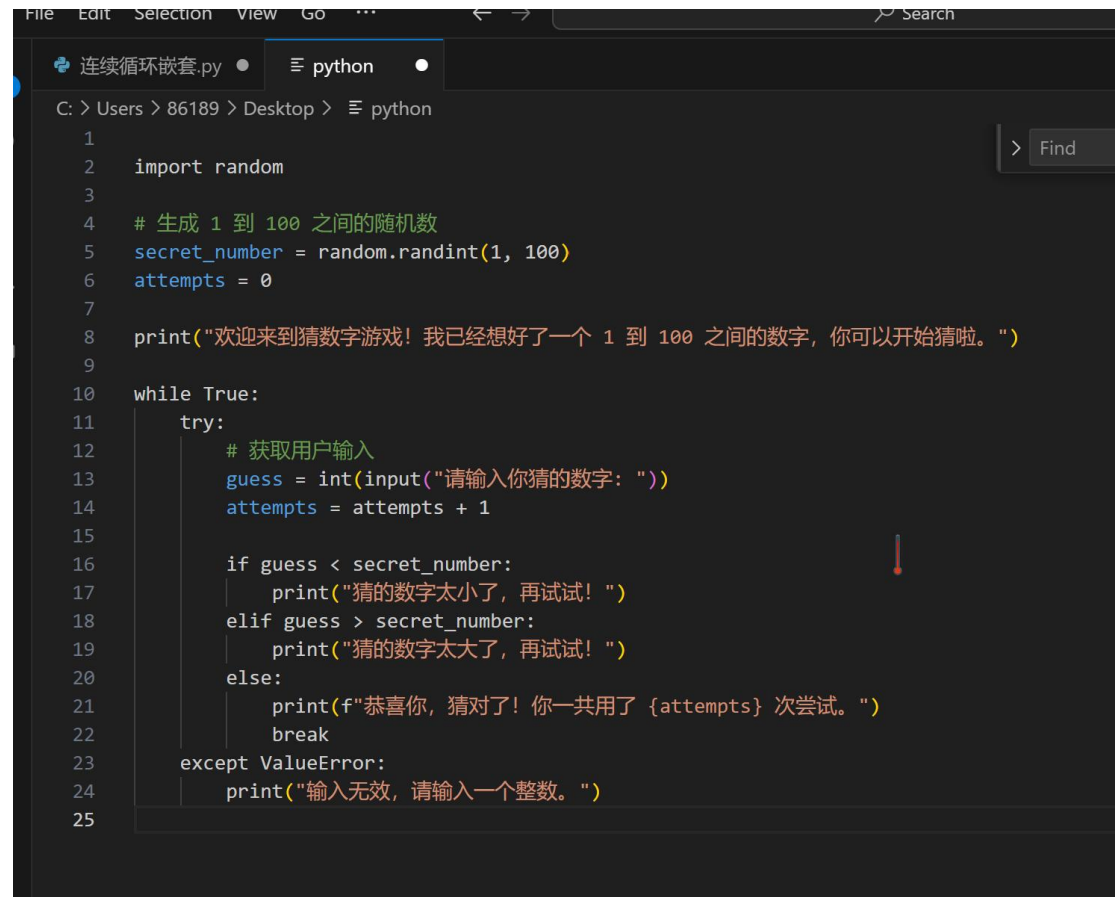


第六周作业

1 猜数字游戏

A screenshot of a Python IDE window. The title bar shows '连续循环嵌套.py' and 'python'. The file path is 'C: > Users > 86189 > Desktop > python'. The code is a number guessing game. It imports the 'random' module, generates a random number between 1 and 100, and initializes 'attempts' to 0. It prints a welcome message. A 'while True' loop starts a 'try' block to get user input, increment attempts, and check if the guess is correct or incorrect, providing feedback. If correct, it breaks the loop. An 'except ValueError' block handles invalid input by asking for an integer.

```
1
2 import random
3
4 # 生成 1 到 100 之间的随机数
5 secret_number = random.randint(1, 100)
6 attempts = 0
7
8 print("欢迎来到猜数字游戏! 我已经想好了一个 1 到 100 之间的数字, 你可以开始猜啦。")
9
10 while True:
11     try:
12         # 获取用户输入
13         guess = int(input("请输入你猜的数字: "))
14         attempts = attempts + 1
15
16         if guess < secret_number:
17             print("猜的数字太小了, 再试试!")
18         elif guess > secret_number:
19             print("猜的数字太大了, 再试试!")
20         else:
21             print(f"恭喜你, 猜对了! 你一共用了 {attempts} 次尝试。")
22             break
23     except ValueError:
24         print("输入无效, 请输入一个整数。")
25
```

2 循环嵌套

for 迭代循环

```
print("for 迭代循环示例:")
```

```
fruits = ["apple", "banana", "cherry"]
```

```
for fruit in fruits:
```

```
    print(fruit)
```

while 条件循环

```
print("\nwhile 条件循环示例:")
```

```
count = 0
```

```
while count < 3:
```

```
    print(f"Count is {count}")
```

```
    count = count + 1
```

break 打断跳出循环

```
print("\nbreak 打断跳出循环示例:")
```

```
numbers = [1, 2, 3, 4, 5]
```

```
for num in numbers:
```

```
    if num == 3:
```

```

        break
    print(num)

# continue 跳至下一轮循环
print("\ncontinue 跳至下一轮循环示例:")
for num in numbers:
    if num == 3:
        continue
    print(num)

# for...else 循环未被打断的处理
print("\nfor...else 循环未被打断的处理示例:")
for num in numbers:
    if num == 6:
        break
else:
    print("循环未被 break 打断")

# if 条件分支
print("\nif 条件分支示例:")
x = 10
if x > 5:
    print("x 大于 5")

# if...elif...elif 多重条件分支
print("\nif...elif 多重条件分支示例:")
y = 2
if y < 0:
    print("y 是负数")
elif y == 0:
    print("y 等于 0")
else:
    print("y 是正数")

# if...else 未满足条件的处理
print("\nif...else 未满足条件的处理示例:")
z = 3
if z > 5:
    print("z 大于 5")
else:
    print("z 不大于 5")

# try...except[...except...else...finally] 捕捉异常的处理
print("\ntry...except 捕捉异常的处理示例:")

```

```
try:
    result = 1 / 0
except ZeroDivisionError:
    print("除数不能为零")
else:
    print("没有发生异常")
finally:
    print("无论是否发生异常，此代码块都会执行")
```

```
# raise 主动抛出异常
print("\nraise 主动抛出异常示例:")
def check_age(age):
    if age < 0:
        raise ValueError("年龄不能为负数")
    return age
```

```
try:
    age = check_age(-5)
except ValueError as e:
    print(e)
```