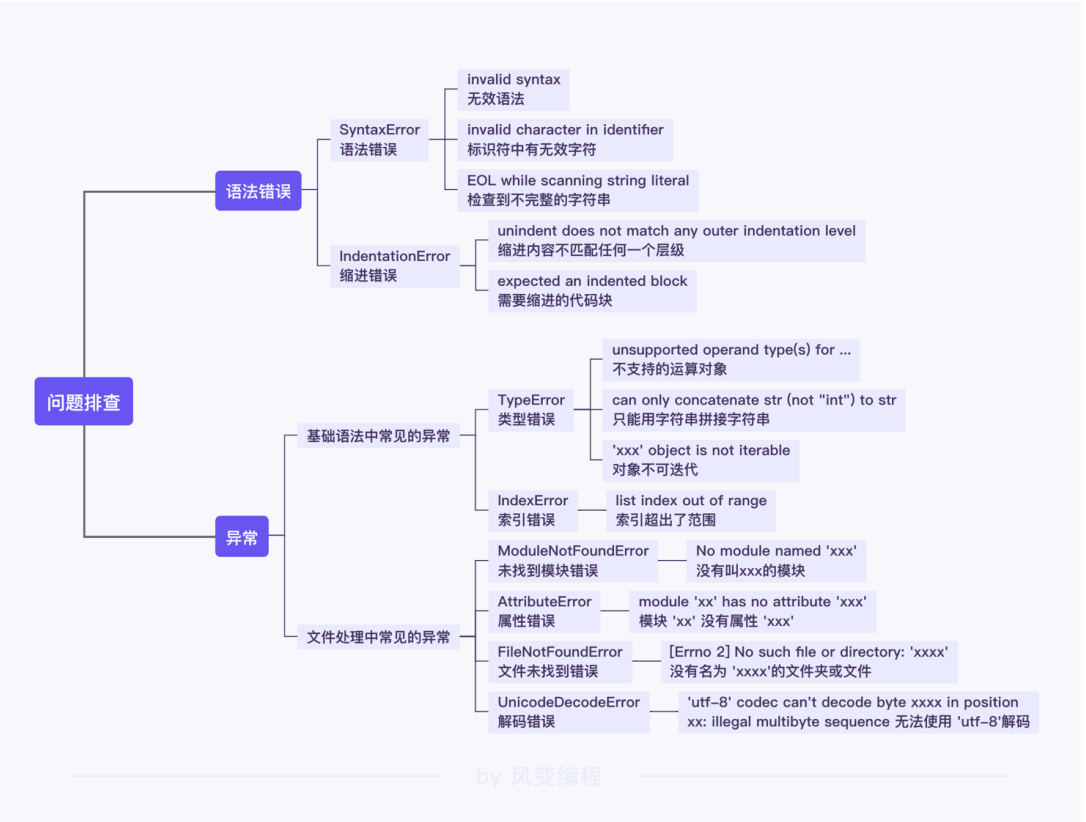


# 基础语法第9关笔记



## 一、常见的语法错误

语法错误(syntax errors)是初学者最容易犯的错误，简单来说就是代码不符合 Python 的基本语法规则而导致程序出了问题。

### 1.SyntaxError

SyntaxError: invalid syntax (无效语法)

已跳过课堂练习

重做

main210.py

```
1 # class 是 Python 的关键字，不可充当变量名
2 class = False
3
4 # True 是 Python 关键字，不可作为函数名
5 def True():
6     print('好好努力，争取offer')
```

终端

```
bash:root$ python /home/python-class/root
/main210.py
File "/home/python-class/root/main210.p
y", line 2
    class = False
    ^
SyntaxError: invalid syntax
```

语法错误中最常见的一种了，通常是由下面几种情况引起的：

1) 遗漏了标点符号；

- 2) 关键字拼写错误或遗漏；
- 3) 变量名或函数名使用了关键字。

### SyntaxError: invalid character in identifier (标识符中有无效字符)



```
main210.py
1 member_list = ['月娜', '廖雨', '王晴', '陈知
2 def member_printer():
3     print(member_list)
```

```
终端
File "/home/python-class/root/main210.p
y", line 1
    member_list = ['月娜', '廖雨', '王晴'
, '陈知枫']
          ^
SyntaxError: invalid character in identif
ier
[]
❌ 第1行出现错误
```

通常是使用了大量中文符号，比如中文括号，中文逗号，中文冒号和中文引号。



```
1 member_list = ['月娜', '廖雨', '王晴', '陈知枫']
2 def member_printer():
3     print(member_list)
```

中文引号，中文逗号

中文括号，中文冒号

中文括号

by 风变编程

### SyntaxError: EOL while scanning string literal (检查到不完整的字符串)



```
main.py
1 print('月娜是老师)
2 print('陈知枫是学生")
```

```
终端
t/main.py
File "/home/python-class/root/main.py"
, line 1
    print('月娜是老师)
          ^
SyntaxError: EOL while scanning string l
iteral
[]
```

通常是因为遗漏了字符串两边的引号，或者混用了引号（即字符串两边的引号不统一）。

### 小结

## SyntaxError 语法错误

### 1. SyntaxError:

invalid syntax 无效语法

- 遗漏了标点符号
- 关键字拼写错误或遗漏
- 变量名或函数名使用了关键字

### 2. SyntaxError:

invalid character in identifier 标识符中有无效字符

- 使用了中文符号

### 3. SyntaxError:

EOL while scanning string literal 检查到不完整的字符串

- 遗漏了字符串两边的引号
- 字符串两边的引号不统一

by 风变编程

## 2.IndentationError

IndentationError: expected an indented block (需要缩进的代码块)



通常是应该缩进的地方没有缩进（例如：if 条件判断，for 循环，while 循环，def 函数定义，class 类的创建）

IndentationError: unindent does not match any outer indentation level (缩进内容不匹配任何一个层级)

已跳过课堂练习

重做

main212.py

```
1 a = 0
2 while a < 5:
3     if a == 3:
4         break
5     a += 1
6     print(a)
```

终端

```
/main212.py
File "/home/python-class/root/main212.py", line 6
    print(a)
    ^
IndentationError: unindent does not match
any outer indentation level
```

通常是缩进滥用，无法确认代码所在的层级，需要跟同层级的代码对齐缩进。

## 小结

### IndentationError 缩进错误

#### 1. IndentationError:

**expected an indented block 需要缩进的代码块**

- 理清代码的执行顺序和逻辑
- 在应该缩进的地方加入缩进

#### 2. IndentationError :

**unindent does not match any outer indentation level 缩进内容不匹配任何一个层级**

- 理清代码块的层级

by 风变编程

## 3.自检清单

## syntax errors自检清单

### 1. SyntaxError:

- 检查是否遗漏逗号或冒号
- 检查是否出现拼写错误
- 检查是否遗漏关键字
- 检查是否使用关键字作为变量，函数或类的名称
- 检查是否使用中文符号
- 检查是否遗漏引号或混用引号

### 2. IndentationError:

- 检查是否存在没有缩进的代码块
- 检查是否有缩进内容不匹配任何层级

by 风变编程

## 二、基础语法中常见的异常

当代码完全符合 Python 的语法规则后，就该尝试运行程序了。但在程序运行过程中，还是有可能发生问题，我们把这类问题称为异常(exceptions)。

一般情况下，异常的“错误消息”会显示 `Traceback (most recent call last):`，而语法错误就不会显示。

The image shows two terminal windows side-by-side. The left window, titled '异常' (Exception), shows a `TypeError: unsupported operand type(s) for -: 'str' and 'str'` error. A yellow box highlights the `Traceback (most recent call last):` line, with an arrow pointing to it from the label '异常'. The right window, titled '语法错误' (Syntax Error), shows a `SyntaxError: invalid character in identifier` error. A yellow arrow points from the label '语法错误' to the error message. Both windows show the file path `File "/home/python-class/root/main.py", line 4, in <module>`.

异常

```
终端
bash:root$ python /home/python-class/root/main.py
Traceback (most recent call last):
  File "/home/python-class/root/main.py",
line 4, in <module>
    print(a * (b - c))
TypeError: unsupported operand type(s) for
-: 'str' and 'str'
```

语法错误

```
终端
File "/home/python-class/root/main.py"
, line 1
    member_list = ['月娜', '廖雨', '王晴', '陈知枫']
                ^
SyntaxError: invalid character in identifier
```

by 风变编程

### 1.TypeError

`TypeError: unsupported operand type(s) for ...`（不支持的运算）



通常是在使用了 `*` / `-` 等运算符进行数字运算时，符号两边的数据类型不为数字类型导致的。

### TypeError: can only concatenate str (not "int") to str （只能用字符串拼接字符串）



这个 `TypeError` 是由不恰当的字符串拼接引起的。

### TypeError: 'xxx' object is not iterable （对象不可被迭代）



这是因为代码尝试遍历一个整数类型的数据，但是整数类型不是一个可迭代对象。一定要记住，使用 `for` 循环去遍历一个数据时，一定要确保其是可迭代对象。

目前为止，可迭代对象应该包括：字符串类型，列表，元组和字典。剩下的一些类型，比如：int 类型，float 类型，布尔类型和 None 就不是可迭代对象。

## 小结

### TypeError 类型错误

#### 1. TypeError:

**unsupported operand type(s) for ... 不支持的运算**

- 检查是否将字符串类型和数字类型进行了运算
- 借助int(), float()转换字符串类型的数据

#### 2. TypeError:

**can only concatenate str (not "int") to str 只能用字符串拼接字符串**

- 检查是否有不恰当的字符串拼接
- 使用str()将非字符串数据转换为字符串类型

#### 3. TypeError:

**'xxx' object is not iterable 对象不可被迭代**

- 检查是否为可迭代对象
- 可迭代对象：字符串，列表，元组，字典

by 风变编程

## 2.IndexError

**IndexError: list index out of range (索引超出了范围)**

已跳过课堂练习

重做

main216.py

```
1 member_list = ['月娜', '廖雨', '王晴', '陈知']
2 print(member_list[4])
```

终端

```
bash:root$ python /home/python-class/root/main216.py
Traceback (most recent call last):
  File "/home/python-class/root/main216.py", line 2, in <module>
    print(member_list[4])
IndexError: list index out of range
```

IndexError 这个异常，和列表取值有关。根据报错信息来看，是因为索引取值超出了列表本身的长度，导致无法取值。

## 三、操作文件过程中的常见异常

### 1.ModuleNotFoundError

❗ 已跳过课堂练习

重做

main217.py

↵ ↗

```
1 import QS
2
3 import OS
4
5 import ps
```

终端

```
bash:root$ python /home/python-class/root
/main217.py
Traceback (most recent call last):
  File "/home/python-class/root/main217.p
y", line 1, in <module>
    import QS
ModuleNotFoundError: No module named 'QS'
[]
```

ModuleNotFoundError 未找到模块错误，是一种有关模块导入的异常。

当导入模块名写错时，都会引发异常 `ModuleNotFoundError: No module named 'xx'`(没有名为 xx 的模块)。

### 2.AttributeError

❗ 已跳过课堂练习

重做

main218.py

↵ ↗

```
1 import os
2 path = './'
3 files_list = os.listdir(path)
4 print(files_list)
```

终端

```
/main218.py
Traceback (most recent call last):
  File "/home/python-class/root/main218.p
y", line 3, in <module>
    files_list = os.listdir(path)
AttributeError: module 'os' has no attrib
ute 'dirlist'
[]
```

AttributeError 常出现于导入模块后调用模块的类/函数/变量。

`AttributeError: module 'os' has no attribute 'dirlist'`翻译过来就是 属性错误：模块 'os' 没有属性 'dirlist'。在使用模块中的方法时，需要先检查拼写是否正确，确认拼写无误后，再检查模块或库中是否存在该方法或属性。

至于如何查找这些方法或属性，可以查阅要使用的模块或库的官方文档，或者在百度输入想要使用的方法。

### 3.FileNotFoundError

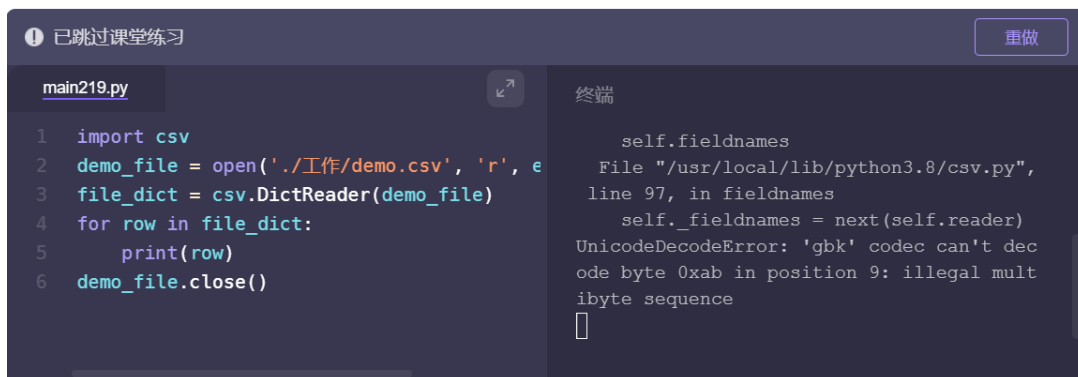




FileNotFoundError 常出现于需要打开文件的操作。

错误消息 `FileNotFoundError: [Errno 2] No such file or directory: './目标文件夹/demo.csv'` 的意思就是：没有文件或文件夹的路径是 `'./目标文件夹/demo.csv'`。

## 4.UnicodeDecodeError



UnicodeDecodeError 常出现于打开文件时 encoding 编码设置。

`UnicodeDecodeError: 'gbk' codec can't decode byte xxxx in position xx: invalid continuation byte`。这就是由编码格式不匹配引发的异常。

遇到这样的异常，你有两种解决方法：

- 第一种，先利用 VSCode 等软件去查看文件的编码格式，然后再将文件的编码格式修改为你在代码中写的编码格式即可。
- 第二种，还是先在 VSCode 里面查看文件的编码格式，然后将代码中的 encoding 参数改为该文件的编码格式即可。

## 小结

## 操作文件过程中的常见异常

### 1. ModuleNotFoundError:

#### 未找到模块错误

- 检查导入了不存在的模块
- 检查模块名拼写是否正确

### 2. AttributeError:

#### 属性错误

- 检查是否使用了模块或库中不存在的方法或属性
- 检查方法或属性是否拼写正确

### 3. FileNotFoundError:

#### 文件未找到错误

- 检查文件路径是否正确

### 4. UnicodeDecodeError:

#### 解码错误

- 检查encoding的编码类型和实际文件是否一致