

目标

- 变量的作用
- 定义变量
- 认识数据类型

一. 变量的作用

图 书 指 南

| | | | | | |
|-----|---|------------------------|------|----|-----------|
| 第一排 | A | 马克思主义、列宁主义、毛泽东思想、邓小平理论 | 第七排 | O | 数理科学和化学 |
| | B | 宗教、哲学 | | P | 天文学、地球科学 |
| 第二排 | C | 社会科学总论 | 第八排 | Q | 生物科学 |
| | D | 政治、法律 | | R | 医药、卫生 |
| 第三排 | E | 军事 | 第九排 | S | 农业科学 |
| | F | 经济 | | T | 工业技术 |
| 第四排 | G | 文化、科学、教育、体育 | 第十排 | U | 交通运输 |
| | H | 语言、文字 | | V | 航空、航天 |
| 第五排 | I | 文学 | 第十一排 | X | 环境科学、安全科学 |
| | J | 艺术 | | Z | 综合类 |
| 第六排 | K | 历史、地理 | 第十二排 | 第十 | 综合类 |
| | N | 自然科学总论 | 第十三排 | 第十 | 期刊 |

举例体验：我们去图书馆读书，怎么样快速找到自己想要的书籍呢？是不是管理员提前将书放到固定位置，并把这个位置进行了编号，我们只需要在图书馆中按照这个编号查找指定的位置就能找到想要的书籍。

这个编号其实就是把书籍存放的书架位置起了一个名字，方便后期查找和使用。

程序中，数据都是临时存储在内存中，为了更快速的查找或使用这个数据，通常我们把这个数据在内存中存储之后定义一个名称，这个名称就是变量。

| | | | | | | | | |
|-----------|-----------|-----------|----|----|-----------|----|-----------|----|
| num1 ↓ | num2 ↓ | num3 ↓ | | | num4 ↓ | | num5 ↓ | |
| 10 | 20 | 30 | 10 | 40 | 20 | 50 | 40 | 10 |
| | | | | | | | | |
| | | | | | | | | |

变量就是一个存储数据的时候当前数据所在的内存地址的名字而已。

二. 定义变量

```
1 变量名 = 值
```

变量名自定义，要满足标识符命名规则。

2.1 标识符

标识符命名规则是Python中定义各种名字的时候的统一规范，具体如下：

- 由数字、字母、下划线组成
- 不能数字开头
- 不能使用内置关键字
- 严格区分大小写

```
1 False      None      True      and        as         assert    break      class
2 continue   def        del        elif       else       except    finally    for
3 from       global    if         import     in         is        lambda     nonlocal
4 not        or         pass       raise      return     try       while      with
5 yield
```

2.2 命名习惯

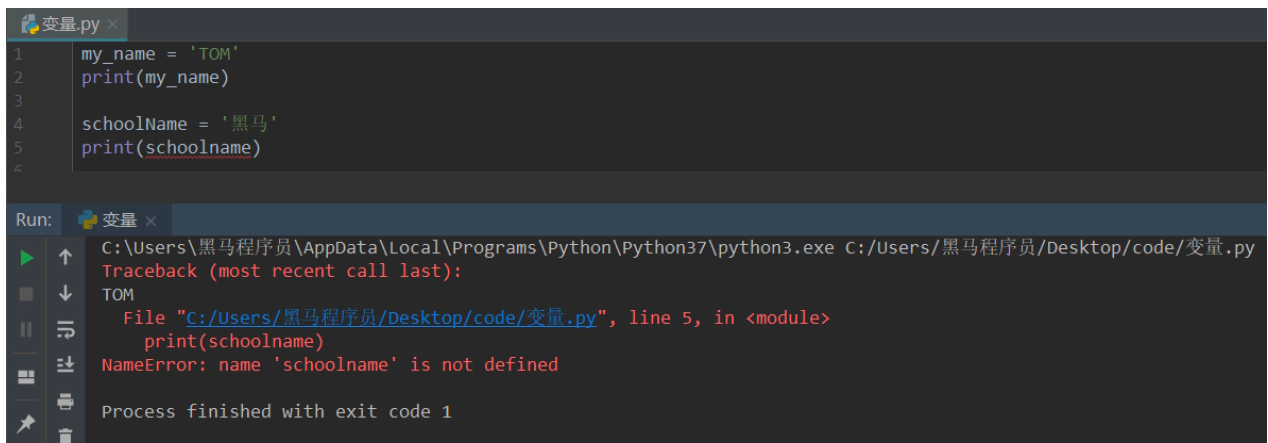
- 见名知义。
- 大驼峰：即每个单词首字母都大写，例如：MyName。
- 小驼峰：第二个（含）以后的单词首字母大写，例如：myName。
- 下划线：例如：my_name。

2.3 使用变量

```
1 my_name = 'TOM'
2 print(my_name)
3
4 schoolName = '黑马程序员'
5 print(schoolName)
```

2.4 认识bug

所谓bug，就是程序中的错误。如果程序有错误，需要程序员排查问题，纠正错误。



The screenshot shows the PyCharm IDE interface. The top pane displays a Python file named '变量.py' with the following code:

```
1 my_name = 'TOM'
2 print(my_name)
3
4 schoolName = '黑马'
5 print(schoolname)
```

The bottom pane shows the 'Run' output. It indicates that the program executed successfully for the first part, printing 'TOM'. However, it then raised a `NameError: name 'schoolname' is not defined` at line 5, where the variable `schoolname` (lowercase) was used instead of `schoolName` (uppercase). The process finished with exit code 1.

三. Debug工具

Debug工具是PyCharm IDE中集成的用来调试程序的工具，在这里程序员可以查看程序的执行细节和流程或者调解bug。

Debug工具使用步骤：

1. 打断点
2. Debug调试

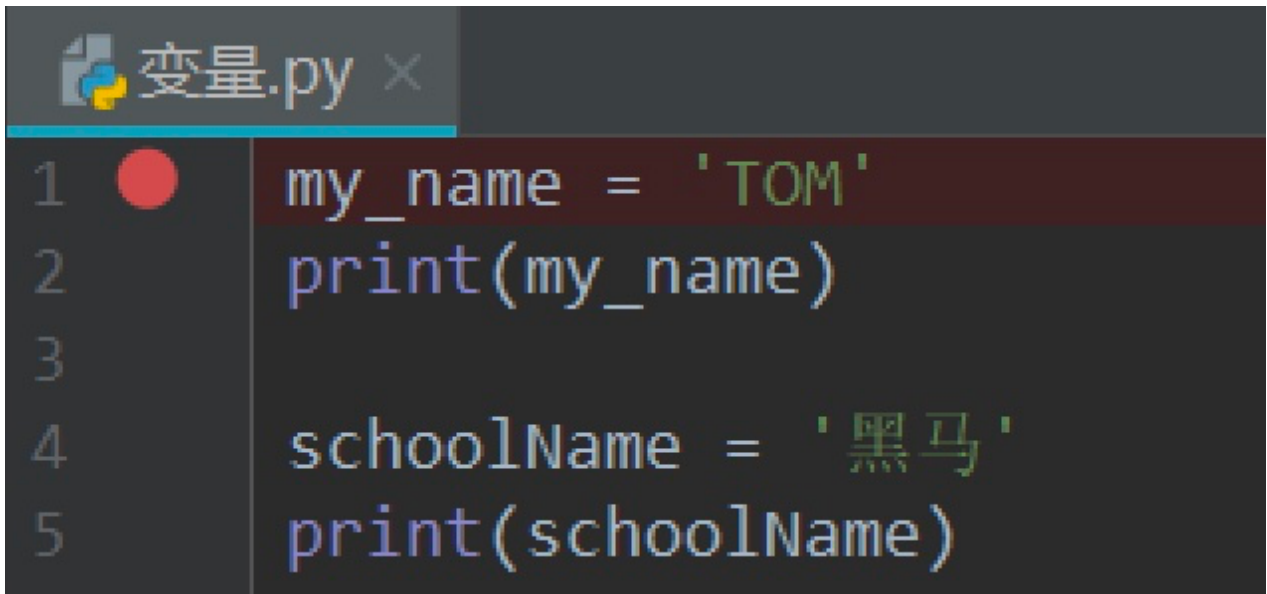
3.1 打断点

- 断点位置

目标要调试的代码块的第一行代码即可，即一个断点即可。

- 打断点的方法

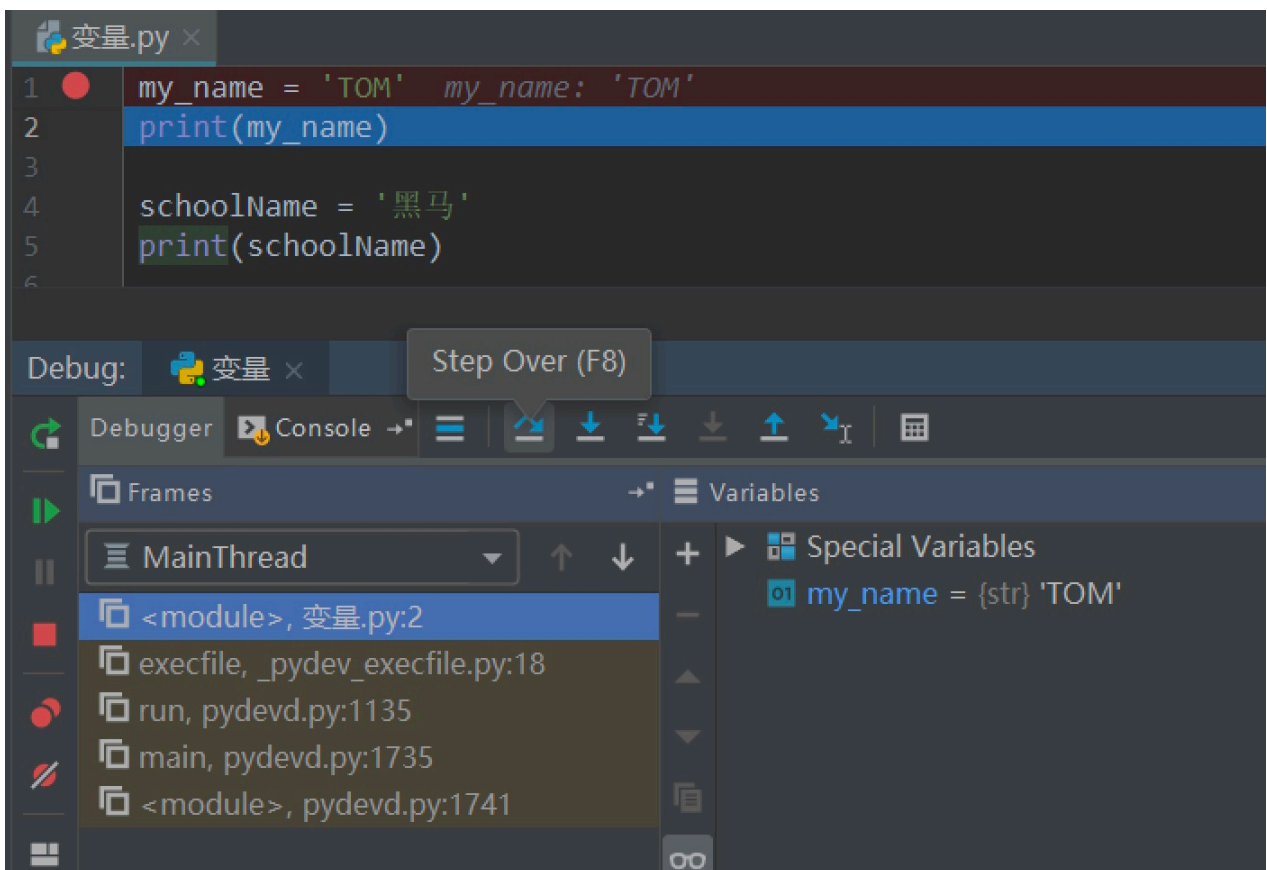
单击目标代码的行号右侧空白位置。



```
1 my_name = 'TOM'
2 print(my_name)
3
4 schoolName = '黑马'
5 print(schoolName)
```

3.2 Debug调试

打成功断点后，在文件内部任意位置 — 右键 -- Debug'文件名' — 即可调出Debug工具面板 -- 单击Step Over/F8，即可按步执行代码。

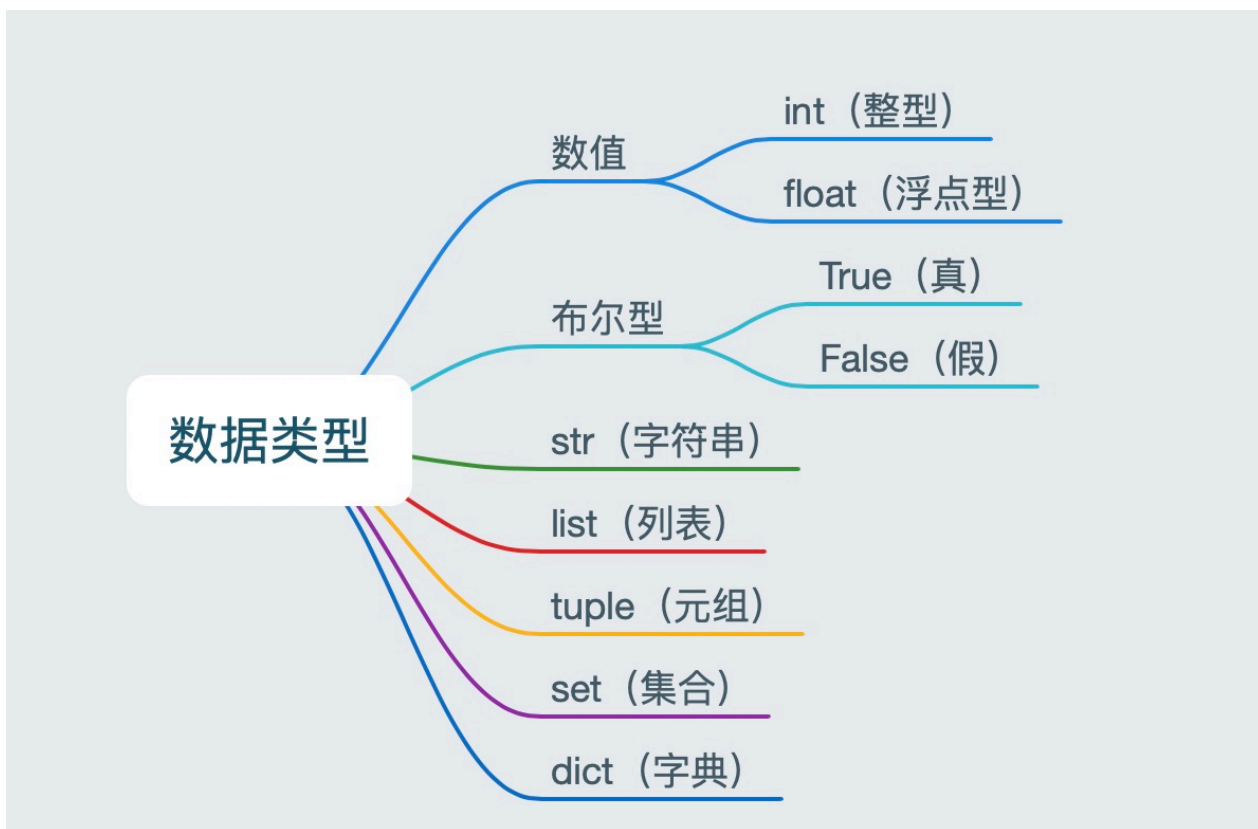


3.2.1 Debug输出面板分类

- Debugger
 - 显示变量和变量的细节
- Console
 - 输出内容

四. 认识数据类型

在 Python 里为了应对不同的业务需求，也把数据分为不同的类型。



检测数据类型的方法：`type()`

```
1 a = 1
2 print(type(a)) # <class 'int'> -- 整型
3
4 b = 1.1
5 print(type(b)) # <class 'float'> -- 浮点型
6
7 c = True
8 print(type(c)) # <class 'bool'> -- 布尔型
9
10 d = '12345'
11 print(type(d)) # <class 'str'> -- 字符串
12
13 e = [10, 20, 30]
14 print(type(e)) # <class 'list'> -- 列表
15
16 f = (10, 20, 30)
17 print(type(f)) # <class 'tuple'> -- 元组
18
19 h = {10, 20, 30}
20 print(type(h)) # <class 'set'> -- 集合
```

```
21  
22 g = {'name': 'TOM', 'age': 20}  
23 print(type(g)) # <class 'dict'> -- 字典
```

总结

- 定义变量的语法

```
1 变量名 = 值
```

- 标识符
 - 由数字、字母、下划线组成
 - 不能数字开头
 - 不能使用内置关键字
 - 严格区分大小写
- 数据类型
 - 整型: int
 - 浮点型: float
 - 字符串: str
 - 布尔型: bool
 - 元组: tuple
 - 集合: set
 - 字典: dict