

别名。

(10) 一般建议每个 import 语句只导入一个模块。

(11) dir()和 help()是两个非常有用的内置函数,前者可以列出指定模块或对象中的成员,后者可以查看相应帮助文档和使用说明。

(12) Python 程序使用缩进来体现代码之间的逻辑关系,并且建议使用必要的空格、空行和注释来提高程序的可读性。

(13) Python 程序中的注释主要有两种形式。

① 以 # 符号开头,表示本行该符号后的所有内容为注释。

② 放在一对三引号之间且不属于任何语句的内容被认为是注释。

(14) 可以使用异常处理结构来提高程序的健壮性,但不建议过多依赖异常处理结构。

(15) 可以通过 Python 脚本的 __name__ 属性来控制脚本程序的某些行为。

(16) Python 程序文件的标准扩展名为 py,Python 支持伪编译将程序转换为字节码,也可以打包为二进制可执行程序。

习 题

1. _____是目前比较常用的 Python 扩展库管理工具。
2. 运算符%_____ (可以、不可以)对浮点数进行求余数操作。
3. 一个数字 5 _____ (是、不是)合法的 Python 表达式。
4. 在 Python 3.x 中 input()函数的返回值类型一律为_____。
5. 简单说明如何选择正确的 Python 版本。
6. 为什么说 Python 采用的是基于值的内存管理模式?
7. 解释 Python 中的运算符/和//的区别。
8. 在 Python 中导入模块中的对象有哪几种方式?
9. 解释 Python 脚本程序的 __name__ 变量及其作用。
10. 编写程序,用户输入一个三位以上的整数,输出其百位以上的数字。例如用户输入 1234,则程序输出 12(提示:使用整除运算)。



素后面加一个逗号“,”。

(5) 将列表、元组或字符串对象与一个整数进行 * 运算,表示将对象中的元素进行重复并返回一个新的同类型对象。

(6) 虽然 + 运算符可以连接两个列表对象,但并不是原地修改列表,而是返回一个新列表,不对原列表对象做任何修改。并且该运算符涉及大量的元素复制操作,效率较低,建议优先考虑使用列表对象的 `append()` 方法。

(7) 推荐使用字典对象的 `get()` 来访问其中的元素。

(8) 列表、字典、集合属于可变序列,元组、字符串属于不可变序列。

(9) 虽然列表支持在列表中间任意位置插入和删除元素,但一般建议尽量从列表的尾部进行元素的增加与删除,这样可以获得更高的速度。

(10) 切片操作不仅可以用来返回列表、元组、字符串中的部分元素,还可以对列表中的元素值进行修改,以及增加或删除列表中的元素。

(11) 关键字 `in` 可以用于列表以及其他可迭代对象,包括元组、字典、`range` 对象、字符串、集合以及迭代器对象,常用在循环语句中对可迭代对象中的元素进行遍历。

(12) 列表推导式可以使用简洁的形式来生成满足特定需要的列表。

(13) 序列解包在多个场合具有重要的应用,是 Python 的基本操作之一。

(14) 字典中的“键”可以是 Python 中任意不可变数据,如整数、实数、复数、字符串、元组等,但不能使用列表、集合、字典作为字典的“键”,因为这些类型的对象是可变的。

(15) 字典中的“键”不允许重复,“值”是可以重复的。

(16) 集合中的所有元素不允许重复,可以使用集合快速提取其他序列中的唯一元素。

(17) 内置函数 `len()` 可以用来返回列表中的元素个数,同样适用于元组、字典、集合、字符串、`range` 对象。

(18) 内置函数 `zip(*iterables)` 可以将多个可迭代对象中对应位置的元素组合为元组,并返回包含这些元组的 `zip` 对象。

(19) 内置函数 `enumerate()` 可以用来枚举列表、元组或其他可迭代对象的元素,返回枚举对象,枚举对象中每个元素是包含下标和元素值的元组。

习 题

1. Python 3.x 的 `range()` 函数返回一个_____。
2. 表达式“`[3] in [1,2,3,4]`”的值为_____。
3. 列表对象的 `sort()` 方法用来对列表元素进行原地排序,该函数返回值为_____。
4. 列表对象的_____方法删除首次出现的指定元素,如果列表中不存在要删除的元素,则抛出异常。
5. 假设列表对象 `aList` 的值为 `[3,4,5,6,7,9,11,13,15,17]`,那么切片 `aList[3:7]` 得到的值是_____。
6. 在 Python 中,字典和集合都是用一对_____作为界定符,字典的每个元素由两部分组成,即_____和_____,其中_____不允许重复。
7. 使用字典对象的_____方法可以返回字典的“键:值”对,使用字典对象



的_____方法可以返回字典的“键”,使用字典对象的_____方法可以返回字典的“值”。

8. 假设有列表 `a=['name','age','sex']` 和 `b=['Dong',38,'Male']`,使用一条语句将这两个列表的内容转换为字典,并且以列表 `a` 中的元素为“键”,以列表 `b` 中的元素为“值”,这个语句可以写为_____。

9. 假设有一个列表 `a`,现要求从列表 `a` 中每 3 个元素取 1 个,并且将取到的元素组成新的列表 `b`,可以使用语句_____。

10. 使用列表推导式生成包含 10 个数字 5 的列表,语句可以写为_____。

11. _____(可以、不可以)使用 `del` 命令来删除元组中的部分元素。

12. 已知列表 `data` 中有若干字符串,要求编写程序,对 `data` 中的字符串进行过滤,只输出重复字符不超过一半的字符串。

13. 编写程序,使用筛选法获取小于 1000 的所有素数。

14. 为什么应尽量从列表的尾部进行元素的增加与删除操作?

15. 编写程序,生成包含 1000 个 0~100 的随机整数,并统计每个元素的出现次数。

16. 编写程序,用户输入一个列表和 2 个整数作为下标,然后输出列表中介于 2 个下标闭区间之间的元素组成的子列表。例如用户输入 `[1,2,3,4,5,6]` 和 2,5,程序输出 `[3,4,5,6]`。

17. 设计一个字典,并编写程序,用户输入内容作为“键”,然后输出字典中对应的“值”,如果用户输入的“键”不存在,则输出“您输入的键不存在!”。

18. 编写程序,生成包含 20 个随机数的列表,然后将前 10 个元素升序排列,后 10 个元素降序排列,并输出结果。



(8) 对用户输入的字符串进行 `eval()` 操作时可能会有安全漏洞,应首先对用户输入的内容进行必要的检查和过滤。

(9) 在 `string` 模块中定义了多个字符串常量,包括数字字符、标点符号、英文字母、大写字母、小写字母等。

(10) 正则表达式是字符串处理的有力工具和技术,可以快速实现字符串的复杂处理任务。

(11) 可以直接使用 `re` 模块的函数来进行字符串处理,也可以将模式编译为正则表达式对象,然后使用正则表达式对象的方法来处理字符串。

(12) 正则表达式中的子模式是作为一个整体来对待的,使用子模式扩展语法可以实现更加复杂的字符串处理任务。

(13) 正则表达式对象的 `match(string[,pos[,endpos]])` 方法用于在字符串开头或指定位置进行搜索,模式必须出现在字符串开头或指定位置;`search(string[,pos[,endpos]])` 方法用于在整个字符串或指定范围中进行搜索:若匹配成功,这两个方法都返回 `Match` 对象,`Match` 对象的主要方法有 `group()`、`groups()`、`groupdict()`、`start()`、`end()`、`span()` 等。

(14) 正则表达式对象的 `findall(string[,pos[,endpos]])` 方法用于在字符串中查找所有符合正则表达式的字符串并以列表形式返回。如果正则表达式中有子模式则只返回子模式匹配的内容,`re` 模块的同名函数也具有这个特点。

习 题

1. 假设有一段英文,其中有单独的字母 `I` 误写为 `i`,请编写程序进行纠正。
2. 假设有一段英文,其中有单词中间的字母 `i` 误写为 `I`,请编写程序进行纠正。
3. 有一段英文文本,其中有单词连续重复了 2 次,编写程序检查重复的单词并只保留一个。例如,文本内容为 "This is is a desk.", 程序输出为 "This is a desk."。
4. 编写程序,用户输入一段英文,然后输出这段英文中所有长度为 3 个字母的单词。



值参数后面不能出现位置参数。

(10) 多次调用函数并且不为默认值参数传递值时,默认值参数只在定义函数时进行解释,对于列表、字典这样可变类型的默认值参数,这一点可能会导致很严重的逻辑错误。

(11) 传递参数时可以使用关键参数,避免牢记参数顺序的麻烦。

(12) 定义函数时,形参前面加一个星号表示可以接收多个位置实参并将其放置到一个元组中,形参前面加两个星号表示可以接收多个关键参数并将其放置到字典中。

(13) 为含有多个形参的函数传递参数时,可以使用 Python 列表、元组、集合、字典以及其他可迭代对象作为实参,并在实参名称前加一个星号,Python 解释器将自动解包,然后传递给多个形参。

(14) lambda 表达式可以用来创建只包含一个表达式的匿名函数。

(15) 在 lambda 表达式中可以调用其他函数,并支持默认值参数和关键参数。

(16) 定义函数时不需要指定其返回值的类型,而是由 return 语句来决定,如果函数中没有执行 return 语句或执行了不返回任何值的 return 语句,则认为该函数返回空值 None。

(17) 在函数内定义的普通变量只在该函数内起作用,称为局部变量。当函数运行结束后,在该函数内部定义的局部变量被自动删除。在函数内部定义的全局变量当函数结束以后仍然存在并且可以访问。

(18) 在函数内部可以通过 global 关键字来声明或者定义全局变量。

(19) Python 3.8 之后,自定义函数单个斜线作为参数表示前面的参数必须为位置参数,单个星号作为参数表示后面的参数必须为关键参数。

习 题

1. 在函数内部可以通过关键字_____来定义全局变量。
2. 如果函数中没有 return 语句或者 return 语句不带任何返回值,那么该函数的返回值为_____。
3. 包含_____语句的函数可以用来创建生成器。
4. 调用带有默认值参数的函数时,不能为默认值参数传递任何值,必须使用函数定义时设置的默认值(判断对错)。
5. lambda 表达式只能用来创建匿名函数,不能为这样的函数起名字(判断对错)。
6. 运行 5.3.1 节倒数第 2、3 段的示例代码,查看结果并分析原因。
7. 编写函数,判断一个整数是否为素数,并编写主程序调用该函数。
8. 编写函数,接收一个字符串,分别统计大写字母、小写字母、数字、其他字符的个数,并以元组的形式返回结果。
9. 在 Python 程序中,局部变量会隐藏同名的全局变量吗? 编写代码进行验证。
10. 编写函数,可以接收任意多个整数并输出其中的最大值和所有整数之和。
11. 编写函数,模拟内置函数 sum()。
12. 编写函数,模拟内置函数 sorted()。
13. 编写函数,模拟内置函数 map()。
14. 编写函数,模拟内置函数 filter()。
15. 使用暴力测试的方法重做例 5-14。




```
d = SubClassD()
d.show()
print(SubClassD.mro())
```

运行结果如下,本书配套 PPT 和微信公众号“Python 小屋”提供了更多 MRO 的示例。

```
Enter SubClassD
Enter SubClassA
Enter SubClassB
Enter SubClassC
BaseClass
Exit SubClassC
Exit SubClassB
Exit SubClassA
Exit SubClassD
[<class '__main__.SubClassD'>, <class '__main__.SubClassA'>, <class '__main__.SubClassB'>, <class '__main__.SubClassC'>, <class '__main__.BaseClass'>, <class 'object'>]
```

本章小结

(1) 面向对象程序设计(Object Oriented Programming, OOP)的思想主要针对大型软件设计而提出,使得软件设计更加灵活,能够很好地支持代码复用和设计复用,并且使得代码具有更好的可读性和可扩展性。

(2) 定义类时使用关键字 class。

(3) 可以动态地为类和对象增加成员。

(4) 类中所有实例方法都至少包含一个 self 参数,并且必须是第一个参数,用来表示对象本身,通过对象名调用实例方法时不需要为 self 参数传递任何值。

(5) 实例属性一般是指在构造方法 __init__() 中定义的,定义时以 self 作为前缀;类属性是在类中所有方法之外定义的数据成员。

(6) 如果通过类名来调用属于对象的公有方法,需要显式为该方法的 self 参数传递一个对象名,用来明确指定访问哪个对象的数据成员。

(7) 在 Python 中,运算符重载是通过重新实现一些特殊函数来实现的。

(8) Python 支持多继承,如果多个父类中有相同名字的成员,而在子类中使用该成员时没有指定其所属父类名,则 Python 解释器将从左向右按顺序进行搜索。

(9) 在 Python 中,以下画线开头的变量名有特殊的含义,尤其是在类的定义中。

(10) 在 IDLE 交互式环境中,单个下画线表示上次语句正常执行的输出结果。

习 题

1. 面向对象程序设计的三要素分别为_____、_____和_____。
2. 与运算符“*”对应的特殊方法名为_____,与运算符“//”对应的特殊方法名为_____。



3. 假设 a 为类 A 的对象且包含一个私有数据成员 __value, 那么在类的外部通过对象 a 直接将其私有数据成员 __value 的值设置为 3 的语句可以写作_____。

4. 继承 6.5 节例 6-4 中的 Person 类生成 Student 类, 编写新的函数用来设置学生专业, 然后生成该类对象并显示信息。

5. 设计一个三维向量类, 并实现向量的加法、减法以及向量与标量的乘法和除法运算。

6. 简单解释 Python 中以下画线开头的变量名的特点。

7. 编写自定义类, 模拟内置集合类。

8. 编写自定义类, 模拟双端队列。

