

习题1: Given the following assignment: `items = ((3, 2), (5, 7), (1, 9), 0, (1))` without writing a Python program or using the Python interpreter, answer the following:

1. What value is returned by `len (items)`?
2. What is the value of `items[1]`?
3. `print items[2][0]` prints 1 but `print items[3][0]` causes a `TypeError`. Why is this given that `items` has more than 3 elements?
4. For `items[x][y]`, what values of `x` and `y` cause the expression to evaluating to 9?
5. It might at first appear that the type of `items[0]` and the type of `items[4]` are the same. (a) What are the types of these two expressions? (b) Why is the type of `items[4]` not tuple? (c) If it was intended that `items[4]` be a tuple, how should the assignment be altered, without adding any new values?

习题2:

Which of the following are types mutable and which are immutable: `int`, `list`, `float`, `tuple`, `str`.

All the types listed are mutable except for `tuple` and `str`.

习题3: What are the types of variables `a` and `b` in the following code: `a=(1) b=(1,)`

习题4: What is the types and value of `y` after executing the following code: `x=[1,2,3,4,5] y=x[:]` Both `x` and `y` are lists. They have the same value, which is `[1, 2, 3, 4, 5]`.

习题5: Examine the following code: `a = 10 b = 20 print "Before swapping a=%d, b=%d" % (a , b) #Swap values a=b b=a print "After swapping a=%d, b=%d" % (a , b)`

1. Try running this small program. What are the values of `a` and `b` after the swap?
2. Why is the result not `a` having the value 20 and `b` having the value 10 as appears to be the intention from the use of the word `swap` in the code?
3. Modify the code so that the final result is `a` having the value 20 and `b` having the value 10 using simultaneous assignment to swap the values.

4. Many languages, such as C, C++, Java, do not have simultaneous assignment. Modify the program so that it successfully swaps the two values without using simultaneous assignment.
5. Both a and b have the value 20.
6. Because the line a=b sets a to 20, leaving b as it was.
7. a,b=b,a
8. a = 10 b = 20 print "Before swapping a=%d, b=%d" % (a, b) # Swap values
temp = a a=b b = temp print "After swapping a=%d, b=%d" % (a, b)

习题6：与python对象有关的三个属性是什么？请简要的描述

习题7：内建函数type()是干什么的？type()返回的对象是什么？

习题8：内建函数str()和repr()之间有什么不同。

习题9：对象相等 type(a) == type(b) 和type(a) is type(b)之间有啥不同？

习题10：

dir()函数作用。试着<http://docs.python.org/2/library/>中随便找一个模块，先import模块，再熟悉dir()的使用。

习题11：以下语句 a = 10 b = 10 c = 100 d = 100 e = 10.0 f = 10.0 请问下面的表达式输出的值是： a is b c is d e is f id(a) id(b) id(c) id(e) id(f)

习题12: 在字符串模块中是不是有一个函数或方法可以鉴定一个字符串是另一个大字符串的一部分。 Strings. Are there any string methods or functions in the string module that will help me determine if a string is part of a larger string?

习题13: 设计一个代码，可以检测字符串是长度，也可以判断字符串长度为一，还可以判断该字符串是不是python关键词 对后一个要求需要用到 import keyword
keyword.kwlist

习题14: 1、用raw_input()函数输入数字，将这些数字用大到小排序。 2、输入字符串，按照字符串的a-z字典顺序排序。 提示：用到while循环和list

习题15: 创建一个string.strip()的替代函数，自己写代码，去掉字符串前后的空格。提示：要注意有多个空格的可能。

习题16: 用列表实现，将一个整型值，返回对应的英文值，比喻89返回“eighty-nine”。假定值是0-1000之间。

习题17: 写一个函数，返回跟一个输入的字符串相似的字符串，要求字符串的大小写反转。比如输入“Mr.ED”，返回“mR.ed”

习题18: 写2个字典，将2个字典合并起来。

习题19: 字典的值可以是任意类型的数据，键只能是特定的数据类型，写出那些数据类型可以作为字典的键。

习题20: 字典和列表的方法： 1、创建一个字典，并把这个字典的键按照字母的顺序显示出来。 2、按照上面的字母排好顺序的键，显示出这个字典中的键和值 3、将字典中的值排好序，再按照排好的序输出键和值。

习题21: 建立字典。有2个长度相同的list，比喻[1,2,3] 和["a","b","c"],用这2个list的数据组成一个这样的字典 {1:"a",2:"b",3:"c"} 提示：用到zip()和dict()

习题22: 颠倒字典中的键和值，将一个字典的键作为另一个字典的值，值作为另一个字典的键。

习题23: 翻译 (a) 编写一个字符翻译程序(功能类似于 Unix 中的 tr 命令)。我们将这个函数叫做 tr(),它有 三个字符串做参数: 源字符串、目的字符串、基本字符串,语法定义如下:

```
def tr(srcstr, dststr, string)
```

srcstr 的内容是你打算“翻译”的字符集合,dststr 是翻译后得到的字符集合,而 string 是你打算进行翻译操作的字符串。举例来说,如果 srcstr == 'abc', dststr == 'mno', string == 'abcdef', 那么 tr()的输出将是'mnodef'. 注意这里 len(srcstr) == len(dststr). 在这个练习里,你可以使用内建函数 chr() 和 ord(), 但它们并不一定是解决这个问题所必不可少的函数。(b) 在这个函数里增加一个标志符参数,来处理不区分大小写的翻译问题。(c)修改你的程序,使它能够处理删除字符的操作。字符串 srcstr 中不能够映射到字符串 dststr 中字符的多余字符都将被过滤掉。换句话说,这些字符没有映射到 dststr 字符串中的任何字符,因此就从函数返回的字符里被过滤掉了。举例来说:如果 srcstr == 'abcdef', dststr == 'mno', string == 'abcdefghi', 那么 tr()将输出'mnoghi'. 注意这里 len(srcstr) >= len(dststr).

习题24: 7-10. 加密。(a) 用上一个练习的思路编写一个"rot13"翻译器。"rot13"是一个古老而又简单的加密方法, 它把字母表中的每个字母用其后的第 13 个字母来代替。字母表中前半部分字母将被映射到后半部分, 而后半部分字母将被映射到前半部分,大小写保持不变。举例来说,'a'将被替换为'n','X'将被替换为'K'; 数字和符号不进行翻译。(b)在你的解决方案的基础上加一个应用程序,让它提示用户输入准备加密的字符串(这个算法 同时也可以对加密后的字符串进行解密),如下所示: % rot13.py Enter string to rot13: This is a short sentence. Your string to en/decrypt was: [This is a short sentence.]. The rot13 string is: [Guvf vf n fubeg fragrapr.].

% rot13.py Enter string to rot13: Guvf vf n fubeg fragrapr. Your string to en/decrypt was: [Guvf vf n fubeg fragrapr.]. The rot13 string is: [This is a short sentence.].

习题25: 随机数。熟读随机数模块然后解下面的题: 生成一个有 N 个元素的由随机数 n 组成的列表, 其中 N 和 n 的取值范围分别为: $(1 < N \leq 100)$, $(0 \leq n \leq 2^{31} - 1)$ 。然后再随机从这个列表中取 N ($1 \leq N \leq 100$) 个随机数 出来, 对它们排序, 然后显示这个子集。

习题26: 修改习题25 的代码: 使用 `random` 模块中的 `randint()` 或 `randrange()` 方法生成一个随机数集合: 从 0 到 9 (包括 9) 中随机选择, 生成 1 到 10 个随机数。这些数字组成集合 A (A 可以是可变集合, 也可以不是)。同理, 按此方法生成集合 B 。每次新生成集合 A 和 B 后, 显示 结果 $A \mid B$ 和 $A \& B$