Python - P38

class set ([iterable]) ，用于创建一个新的集合(set)

class frozenset([iterable])，用于创建一个新的不可变集合(frozenset)

在Python中，set / frozenset 都属于 set type

set object : unordered collection of distinct hasable objects

各不相同的可散列对象组成的无序集合

与其他 collection 类似

支持通过关键字 in 判断对象是否在集合中(membership testing)

支持通过 len() 函数返回集合的大小(cardinality)

支持 直接用于增强 for 循环以遍历集合中的元素

注意：由于 set 是无序的，所以遍历的顺序可能与插入顺序不同

set 与 frozenset 的主要区别在于 mutable / immutable

set 是可变的(mutable)

其内容可以通过 add() / remove() 等方法修改

由于 set 是可变的，所以没有散列值(hash value)

于是不可作为字典的键(dictionary key)

也不可作为其他集合的元素

frozenset 是不可变的(immutable)

其内容在创建后即不可改变(cannot be altered)

由于 frozenset 是不可变的，所以是可散列的(hashable)

于是可用作字典的键或者其他集合的元素

特别地有，集合可以通过字面值的表示方法定义，

如 set = {'a', 'b', 'c'}

集合运算符：在Python中，支持以算术/逻辑/比较运算符表示的集合运算符

set <= other，set.issubset(other)：子集 set ⊆ other

set < other：真子集(proper subset) set ⊂ other

set >= other，set.issuperset(other)：超集 set ⊇ other

set > other：真超集(proper superset) set ⊃ other

set | other | ...，set.union(*others)：并集 set ∪ other ∪ ...

set & other & ...，set.intersection(*others)：交集 set ∩ other ∩ ...

set - other - ...，set.difference(*others)：差集 set \ other \ ...

set ^ other，set.symmetric_difference(other)：对称差 set ⊕ other

set.isdisjoint(other)：判断 set 与 other 是否不相交

Python - P39

集合赋值运算符：在Python中，set [redacted] 支持赋值运算符

set |= other |... ，set.update(*others)：并集并赋值

set &= other &... ，set.intersection_update(*other)：交集并赋值

set -= other |... ，set.difference_update(*other)：差集并赋值

set ^= other，set.symmetric_difference_update(other)：对称差并赋值

集合修改：在Python中，set支持对集合的修改方法

set.add(elem)：向集合set中添加元素elem

set.remove(elem)：从集合中删除元素elem，

如果elem不存在，则抛出KeyError错误

set.discard(elem)：如果elem存在于集合set中，则删除elem

set.pop()：从集合中移除并返回一个任意元素

如果集合set为空，则抛出KeyError错误

set.clear()：从集合中移除所有元素

globals()：返回一个字典，包含当前的所有全局变量

return a dictionary representing current global symbol table

始终是当前模块的字典（dictionary of current module）

如果在函数或方法的内部（inside function or method）

则返回[redacted]定义函数的模块（module where it defined）

而非调用函数的模块（module from which it called）

hash(object)，[redacted]返回对象的散列值（如果散列值存在）

return hash value of object (if it has one)

散列值必定是整型数（integer）

用于在字典检索时快速比较字典的关键字

quickly compare dictionary keys during dictionary lookup

[redacted]相等的数值必定拥有相等的散列值，即使数值类型不同

numeric values compared equal have the same hash value

even of different types，如整数1和浮点值1.0

对于用户定义的__hash__()方法，(custom __hash__() method)

hash()函数会根据本地的字长来压缩返回值

truncate return value based on bit width of host machine

Python - P40

object._hash_(self)，由built-in函数 hash() 调用
用于 hashed collection 中成员上的运算，如 set / frozenset / dict
hash() 函数应返回一个整形值
only required property : objects compared equal have the same hash value
advised to mix together hash values of components of objects
by packing components into tuple and hashing the tuple
如 def _hash_(self):
return hash((self.name, self.age))
注意 hash() 会根据参数 Py_ssize_t 调整用户定义 _hash_() 的返回值
typically 8 byte on 64-bit builds / 4 byte on 32-bit builds
通过 python -c "import sys; print(sys.hash_info.width)" 查询
与 object._eq_(self, other) 方法的关系
如果类没有定义 _eq_()，则也不应定义 _hash_()
如果类定义了 _eq_() 但没有定义 _hash_()
则该不可用于 hashable collection
如果一个类中定义了可变对象 (mutable object) 且实现了 _eq_()
则不应实现 _hash_()
由于 hashable collection 要求 key 的 hash value 是不可变的 (immutable)
否则 object in wrong hash bucket when hash value change
user-defined class has _eq_() and _hash_() by default
all objects compare unequal with others
hash() return appropriate value
x == y implie both x is y and hash(x) == hash(y)
class override _eq_() but not define _hash_()
will have _hash_() implicitly set to None
instance raise appropriate TypeError
when program attempt to retrieve hash value
unhashable when isinstance(obj, collections.abc.Hashable)
如果需要从父类继承 _hash_() 的实现
则需要显式地定义 _hash_ = <ParentClass>._hash_
如果没有 override _eq_() 但希望 禁止支持 hash
则应显式地定义 _hash_ = None
否则定义的 _hash_() 会显式地抛出 TypeError
但错误地被认为 hashable

Python - P41

id (object)，返回对象的 "identity"，为一个整型值 (integer)
guaranteed to be unique and constant for the object during lifetime
two objects with non-overlapping lifetime may have same id() value

input ([prompt])，接收用户的输入并转换为一个字符串并返回
prompt：可选地传入一个字符串，显示在输入之前
written to standard output without trailing newline
read one line from input, convert to string, strip trailing newline, and return
when EOF read, EOFError raised

isinstance (object, classinfo)，判断 object 是否是 classinfo 的对象
return TRUE if object argument is instance of classinfo argument
or of direct/indirect/virtual subclass
classinfo：传入用于判断的 type 信息
可以是单独的 type，也可以是 tuple of type
当传入type的元组时，则只要其中任意一个匹配即返回 TRUE
如果 classinfo 传入既非 type 也非 tuple of type
则抛出异常 TypeError

issubclass (class, classinfo)，判断 class 是否 classinfo 的子类，或者 classinfo 是否 class 的基类
return TRUE if class is direct/indirect/virtual subclass of classinfo
class considered subclass of itself
与 isinstance 类似，classinfo 可是是单独的 type 或 type 的元组
且只要任意一个匹配即返回 TRUE
如果 classinfo 传入既非 type 也非 tuple of type
则抛出 TypeError 异常

len (s)，返回传入对象的长度 (number of items)
参数 s 可以是 sequence (string/bytes/tuple/list/range)
或 collection (dictionary/set/frozenset)

locals ()，返回包含当前局部变量的字典，dictionary representing current local symbol table
free variable returned in function block, not in class block
locals() same as global() at module level