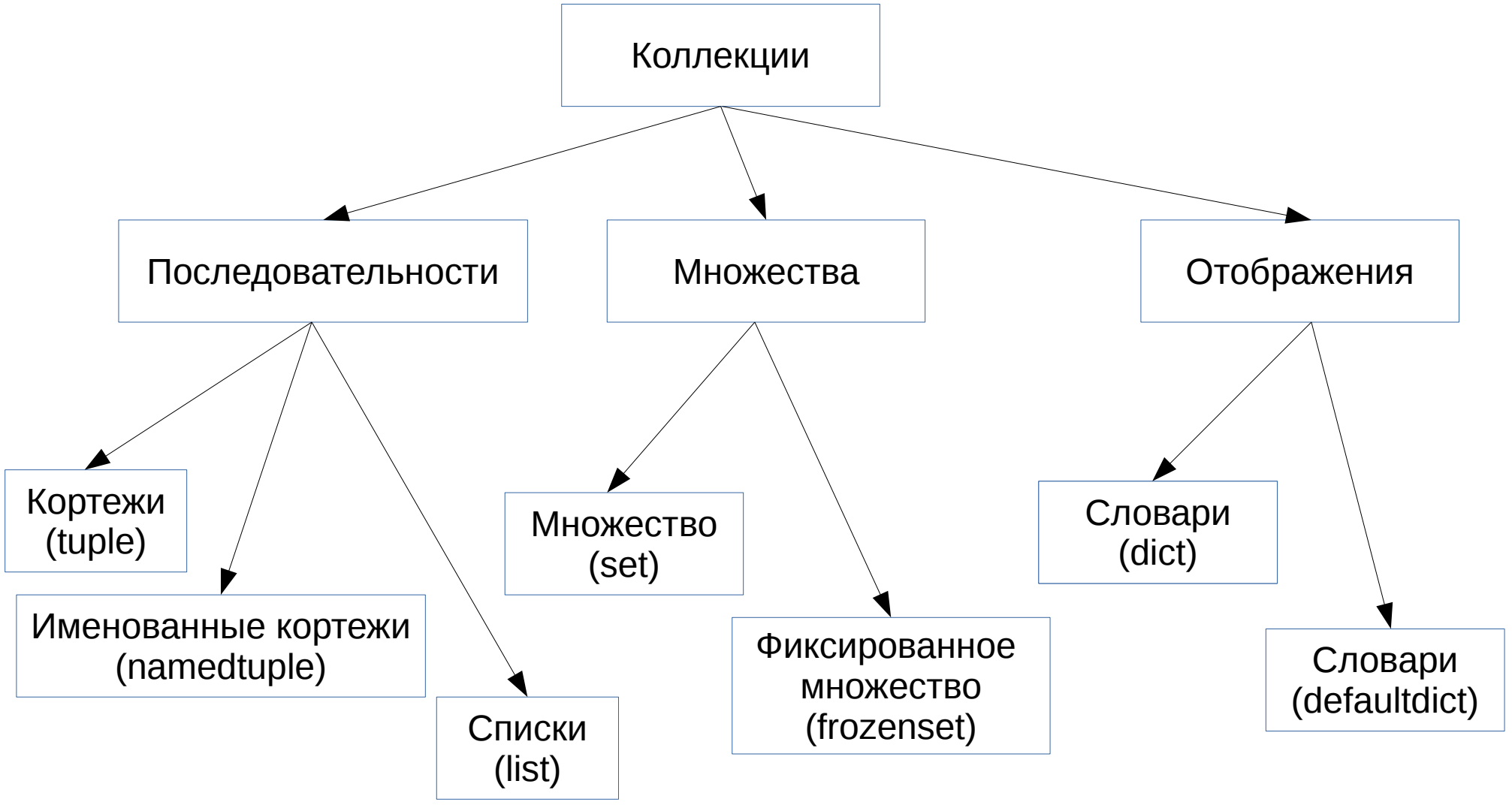

Коллекции
Последовательности
Множества
Отображения
Кортежи
(tuple)
Именованные кортежи
(namedtuple)
Списки
(list)
Множество
(set)
Фиксированное
множество
(frozenset)
Словари
(dict)
Словари
(defaultdict)

Множества - разновидность коллекций объектов, обладающих следующими свойствами:

- элементы множества уникальные (не повторяются).
- элементы множества неизменяемые (хешируемые)
- множества неупорядочены

Области применения множеств

1. Когда нужна быстрая проверка на вхождение в множество
2. Обеспечение уникальности (неповторяемости) элементов
3. Специальные операции над множествами (объединение, пересечение и т.д)

Создание множества

1. Перечислением элементов: a = {1, 2, 3}
2. Создание пустого множества: a = set()
3. Создание множества из другой коллекции:

    a = set('abracadabra')

    a == {'c', 'b', 'a', 'r', 'd'}

    # Повторяющиеся элементы не добавлены в множество
4. С помощью генераторов:

    a = {i for i in range(5)}

Работа с элементами множества

1. Обход элементов циклом for в **неопределенном** порядке:

   for el in a:

2. Проверка на вхождение в множество (**in**, **not in**):

   if 'b' **in** a: # выполняется быстрее чем в списке

3. Добавление в множество:

   a.**add**('n') # в отличии от списка, там append

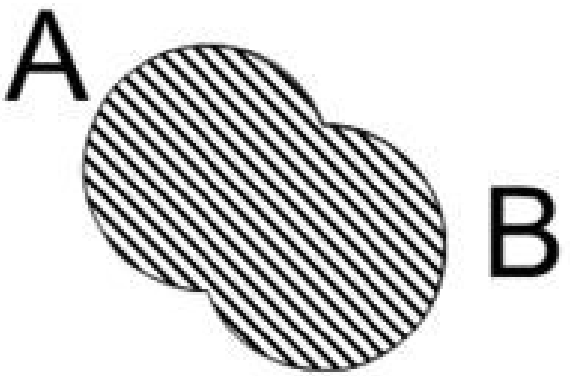
Методы множеств

Имя	Описание
s.clear()	Удаляет все элементы из множества s
s.copy()	Возвращает поверхностную копию множества
s.discard(x)	Удаляет элемент x из множества s, если он присутствует в множестве s
s.remove(x)	Удаляет элемент x из множества s или возбуждает исключение KeyError, если элемент x отсутствует в множестве s
s.isdisjoint(t)	Возвращает True, если множества s и t не имеют общих элементов
s.pop()	Возвращает и удаляет случайный элемент множества s

Операции над множествами

Объединение множеств

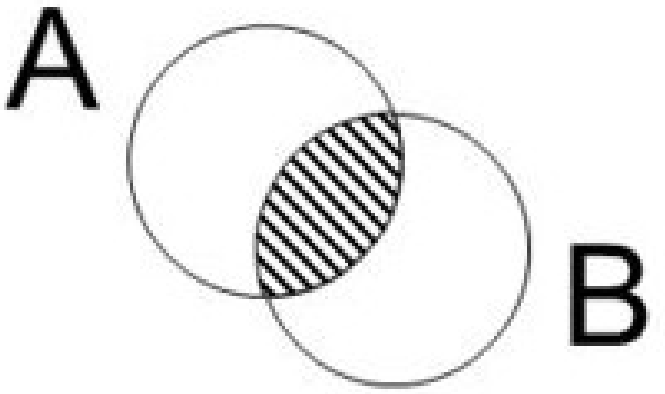
A.union(B)
A | B



set("pecan") | set("pie") == {'p', 'e', 'c', 'a', 'n', 'i'} # Объединение

Пересечение множеств

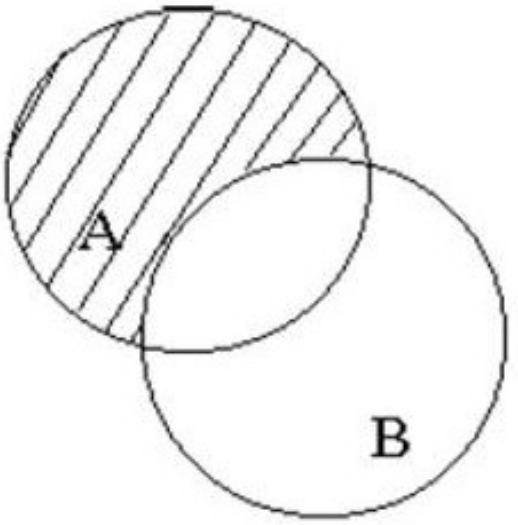
A.intersection(B)
A & B



set("pecan") & set("pie") == {'p', 'e'}

Разность множеств

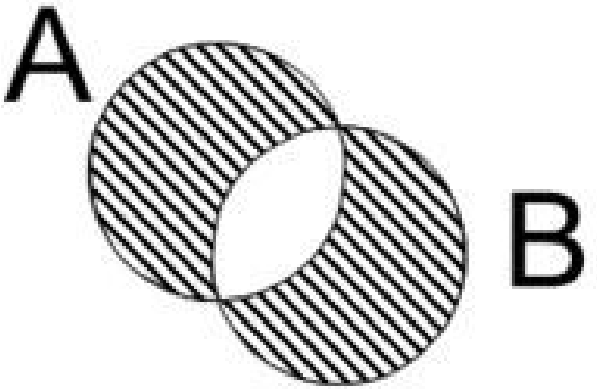
A.difference(B)
A - B



set("pecan") - set("pie") == {'c', 'a', 'n'}

Строгая дизъюнкция (исключающее ИЛИ)

A.symmetric_difference(B)
A ^ B

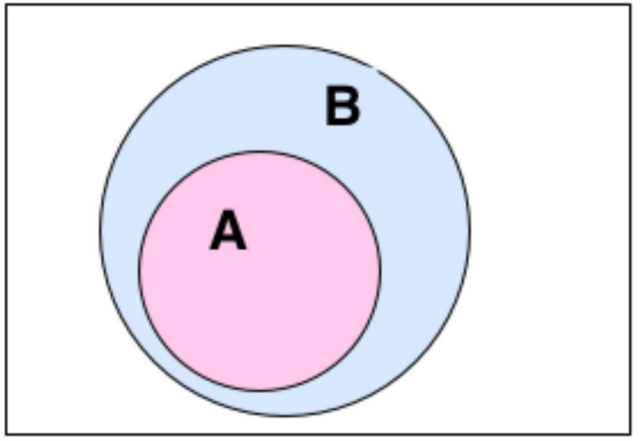


set("pecan") ^ set("pie") == {'c', 'a', 'n', 'i'}

Проверка является ли множество подмножеством другого

A.**issubset**(B)
A <= B

Возвращает True, если множество A эквивалентно множеству B или является его подмножеством;



Проверка является ли множество надмножеством другого

A.**issuperset**(B)
A >= B

Возвращает True, если множество A эквивалентно множеству B или является его надмножеством

