

# Lista de Exercício - Exercício I

Rafael Gonçalves de Oliveira Viana

2º semestre de 2017

1. Qual é a diferença entre um Set, Dicionário e Lista ?. Implemente um exemplo de cada

## **Set**

Um conjunto é uma coleção desordenada de dados, sem elementos duplicados. Usos comuns para isso incluem verificações da existência de objetos em outras sequências e eliminação de itens duplicados. Conjuntos também suportam operações matemáticas como união, interseção, diferença e diferença simétrica.

## **Dicionários**

Dicionários são também chamados de “memória associativa”, ou “vetor associativo”. Diferentemente de sequências que são indexadas por inteiros, dicionários são indexados por chaves (keys), que podem ser de qualquer tipo imutável (como strings e inteiros). Tuplas também podem ser chaves se contiverem apenas strings, inteiros ou outras tuplas. Se a tupla contiver, direta ou indiretamente, qualquer valor mutável não poderá ser chave. Listas não podem ser usadas como chaves porque são mutáveis.

O melhor modelo mental de um dicionário é um conjunto não ordenado de pares chave-valor, onde as chaves são únicas em uma dada instância do dicionário.

Dicionários são delimitados por : . Uma lista de pares chave:valor separada por vírgulas dentro desses delimitadores define a constituição inicial do dicionário. Dessa forma também será impresso o conteúdo de um dicionário em uma seção de depuração.

As principais operações em um dicionário são armazenar e recuperar valores a partir de chaves. Também é possível remover um par chave:valor com o comando `del`. Se você armazenar um valor utilizando uma chave já presente, o antigo valor será substituído pelo novo. Se tentar recuperar um valor dada uma chave inexistente será gerado um erro.

O método `keys()` do dicionário retorna a lista de todas as chaves presentes no dicionário, em ordem arbitrária (se desejar ordená-las basta aplicar o método `sort()` na lista devolvida). Para verificar a existência de uma chave, utilize o método `has_key()` do dicionário ou a keyword `in`.

## **Listas**

A lista é uma estrutura de dados, que pode ser escrita como uma lista de valores separados por vírgula e entre colchetes. Mais importante, os valores contidos na lista não precisam ser do mesmo tipo.

## **Conclusão**

Python possui diversas estruturas de dados nativas utilizadas para agrupar outros valores, cada uma delas pode ser aplicada, em situações diferentes, assim aproveitando todos os recursos disponíveis da estrutura de dado escolhida, porém a mais versátil de todas é sem dúvida a lista.