**Linguagem de Programação**

* **Cw2.1 – Listas, tuplas , set e dicionário.**

**Listas** – estrutura de dados mutável. Ou seja, novos valores podem ser adicionados ou removidos.

A ultima posição é n – 1

Ex: vogais = [‘a’, ‘e’, ‘i’, ‘o’, ‘u’]

A lista pode ser criada sem nenhum elemento e feita posteriormente:

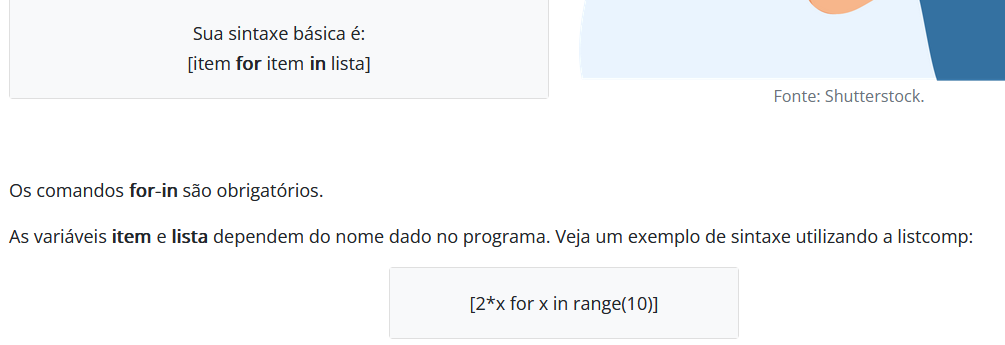
Para acessar o valor guardado em

Uma lista, basta indicar o nome da

Variável e a posição do elemento.

Ex: vogais[2]

**List Comprehension –** também chamada de **listcomp**



**Tuplas** – estrutura de dados de objetos tipo sequencia imutáveis.

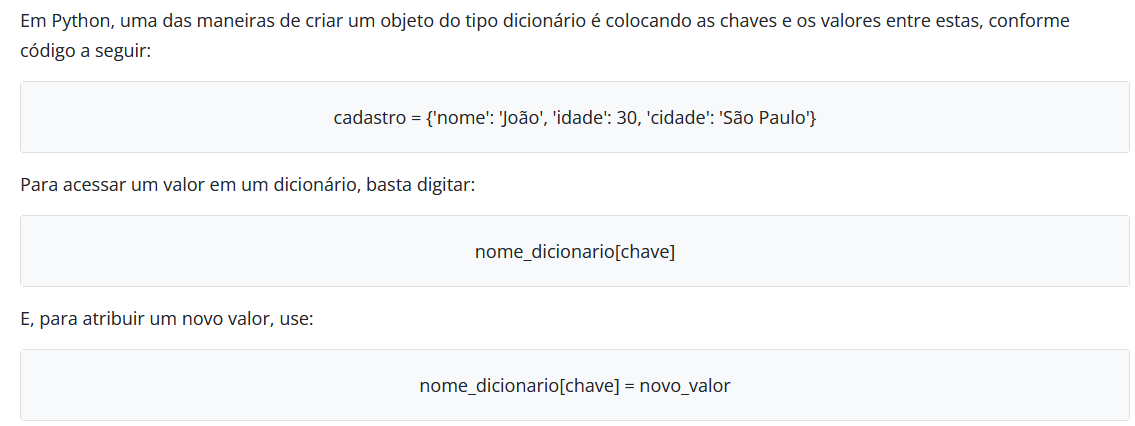
Uma das maneiras de criar uma tupla é colocando valores entre **parênteses**. Ao contrario da lista, uma tupla não permite a inserção posterior de dados, mas os dados podem ser acessados pela sua posição na sequencia.

Ex: vogais = (‘a’, ‘e’, ‘i’, ‘o’, ‘u’)

**Sets** – estrutura para operações matemáticas de conjuntos, tais como: união, intersecção, diferença e etc. Uma das maneiras de se criar um objeto **set** é colocnado os valores entre **chaves**. Um **set** permite a inserção de valores posteriores a sua criação com a função **add()**, mas não permite acessar valores pela sua posição.

Ex: vogais = {‘a’, ‘e’, ‘i’, ‘o’, ‘u’}

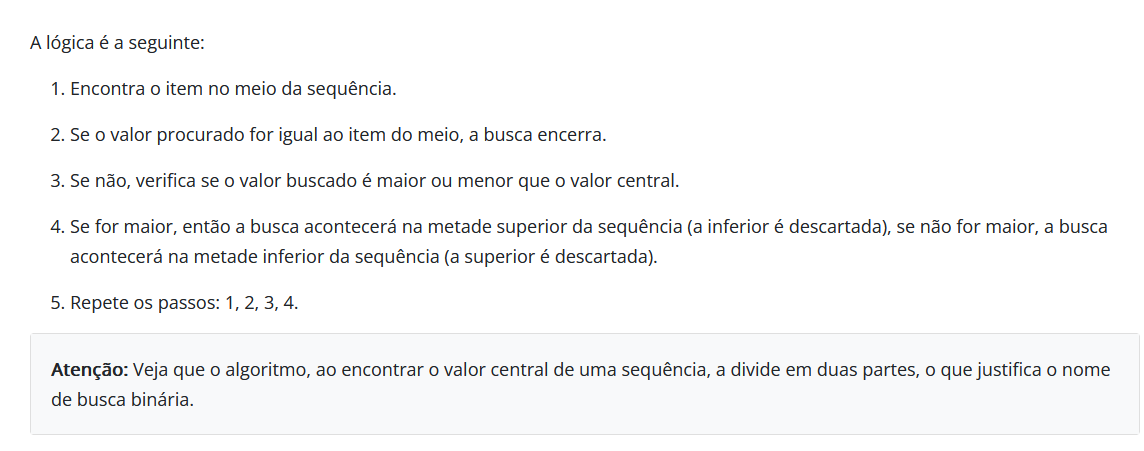
**Dicionários** – estrutura de dados que possuem um mapeamento entre uma chave e um valor são considerados objetos do tipo **mapping.**



* **Cw2.2 – Algoritmos de busca**

Busca **Sequencial –** É necessário percorrer todos os elementos da lista até encontrar o elemento procurado.

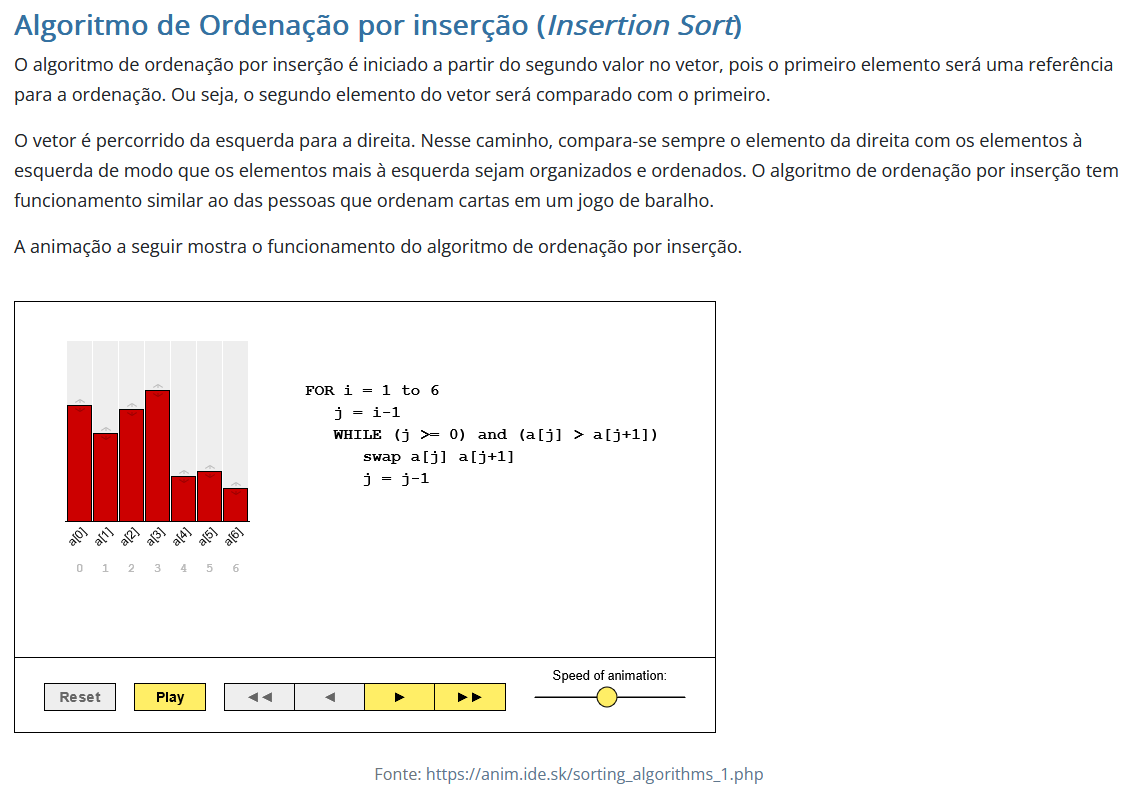
**Algoritmo de busca binária –** A diferença entre algoritmo de busca linear e algoritmo de busca binária é que neste os valores precisam estar ordenados.



* **Cw2.3 – Algoritmos de ordenação**

**Classificação dos algoritmos –**

Complexidade **O(N2).** Nesse grupo estão os algoritmos **selection sort, bubble sort e insertion sort.** São algoritmos lentos para ordenar em grandes listas porém são intuitivos de entender e possuem fácil codagem.

Complexidade **O(N log N)**. Neste grupo, vamos conhecer **merge sort e quick sort.** Possuem performance superior, porém são um pouco mais complexos de serem implementados.

