

FUNÇÕES PARA MANIPULAR DICIONÁRIOS EM PYTHON

COMANDO	O QUE ELE FAZ?	EXEMPLO
.clear() <i>Dicionários e Sets</i>	Remove todas as chaves e os valores do dicionário	dicio = {'um': 1, 'dois': 2, 'três': 3} dicio.clear() dicio = {}
.copy() <i>Dicionários e Sets</i>	Copia as chaves e os valores do dicionário	dicio1 = {'um': 1, 'dois': 2, 'três': 3} dicio2 = dicio1.copy() dicio2 = {'um': 1, 'dois': 2, 'três': 3}
.fromkeys(chave, valor)	Cria um dicionário a partir das chaves informadas	chaves = (1, 2, 3) dicio = dict.fromkeys(chaves, 0) dicio = { 1 : 0, 2 : 0, 3 : 0}
.get(chave)	Retorna o valor atribuído a uma chave	dicio = {'um': 1, 'dois': 2, 'três': 3} chave = dicio.get('dois') chave = 2
.items()	Retorna uma lista contendo tuplas com as chaves e os valores do dicionário	dicio = {'a': 1, 'b': 2, 'c': 3} dicio_itens = dicio.items() dicio_itens = [('a', 1), ('b', 2), ('c', 3)]

COMANDO	O QUE ELE FAZ?	EXEMPLO
.keys()	Retorna uma lista contendo as <u>chaves</u> do dicionário	dicio = {1: 0, 2: 0, 3: 0} chaves = dicio.keys() chaves = dict_keys([1, 2, 3])
.pop(chave) <i>Dicionários e Sets</i>	Remove do dicionário a chave informada e o seu respectivo valor	dicio = {'a': 1, 'b': 2, 'c': 3} dicio.pop('b') dicio = {'a': 1, 'c': 3}
.popitem()	Remove do dicionário a <u>última</u> chave e o seu respectivo valor	dicio = {'a': 1, 'b': 2, 'c': 3} dicio.pop() dicio = {'a': 1, 'b': 2}
.update({chave: valor})	Atualiza a chave e o valor de um dicionário	dicio = {'a': 1, 'b': 2, 'c': 3} dicio.update({'b': 8}) dicio = {'a': 1, 'b': 8, 'c': 3}
.values()	Retorna uma lista contendo os <u>valores</u> do dicionário	dicio = {'a': 1, 'b': 2, 'c': 3} valores = dicio.values() valores = dict_values(['um', 'dois'])

FUNÇÕES PARA MANIPULAR SETS EM PYTHON

COMANDO	O QUE ELE FAZ?	EXEMPLO
<code>.add(valor)</code>	Adiciona um valor ao set informado	<code>set = {1, 2, 3}</code> <code>set.add(4)</code> <code>set = {1, 2, 3, 4}</code>
<code>.remove(valor)</code>	Remove do set o valor informado	<code>set = {1, 2, 3}</code> <code>set.remove(3)</code> <code>set = {1, 2, 4}</code>
<code>.update(set)</code>	Atualiza os valores de um set a partir de outro set	<code>set1 = {1, 2, 3}</code> <code>set2 = {4, 5, 6}</code> <code>set1.update(set2)</code> <code>set1 = {1, 2, 3, 4, 5, 6}</code>
<code>.union(set)</code>	Retorna um set contendo a união de outros dois informados	<code>set1 = {1, 2, 3}</code> <code>set2 = {4, 5, 6}</code> <code>set3 = set1.union(set2)</code> <code>set3 = {1, 2, 3, 4, 5, 6}</code>

COMANDO	O QUE ELE FAZ?	EXEMPLO
.discard(valor)	Remove um item especificado do set	<pre>set1 = {"google", "microsoft", "apple"} set1.discard("microsoft") set1 = {"google", "apple"}</pre>
.difference(set)	Retorna a diferença de um set por outro set	<pre>set1 = {"apple", "banana", "cherry"} set2 = {"google", "microsoft", "apple"} set3 = set1.difference(set2) set3 = {"banana", "cherry"}</pre> <p>ou</p> <pre>set1 = {"apple", "banana", "cherry"} set2 = {"google", "microsoft", "apple"} set3 = set2.difference(set1) set3 = {"google", "microsoft"}</pre>
.items()	Retorna uma lista contendo tuplas com as chaves e os valores do dicionário	<pre>dicio = {'a': 1, 'b': 2, 'c': 3} dicio_itens = dicio.items() dicio_itens = [('a', 1), ('b', 2), ('c', 3)]</pre>