

# Python - Objetos de visualização de dicionário

Os métodos `items()`, `chaves()` e `valores()` da classe `dict` retornam objetos de visualização. Essas visualizações são atualizadas dinamicamente sempre que ocorre alguma alteração no conteúdo do objeto do dicionário de origem.

## Método de itens()

O método `items()` retorna um objeto de visualização `dict_items`. Ele contém uma lista de tuplas, cada tupla composta pelos respectivos pares de chave e valor.

## Sintaxe

```
Obj = dict.items()
```

## Valor de retorno

O método `items()` retorna o objeto `dict_items` que é uma visão dinâmica de tuplas (chave, valor).

## Exemplo

No exemplo a seguir, primeiro obtemos o objeto `dict_items()` com o método `items()` e verificamos como ele é atualizado dinamicamente quando o objeto do dicionário é atualizado.

```
numbers = {10:"Ten", 20:"Twenty", 30:"Thirty",40:"Forty"}
obj = numbers.items()
print ('type of obj: ', type(obj))
print (obj)
print ("update numbers dictionary")
numbers.update({50:"Fifty"})
print ("View automatically updated")
print (obj)
```

Ele produzirá a seguinte **saída** -

```
type of obj: <class 'dict_items'>
dict_items([(10, 'Ten'), (20, 'Twenty'), (30, 'Thirty'), (40, 'Forty')])
update numbers dictionary
```



View automatically updated

```
dict_items([(10, 'Ten'), (20, 'Twenty'), (30, 'Thirty'), (40, 'Forty'), (50, 'Fifty')])
```

## Método chaves()

O método `keys()` da classe `dict` retorna o objeto `dict_keys` que é uma lista de todas as chaves definidas no dicionário. É um objeto de visualização, pois é atualizado automaticamente sempre que qualquer ação de atualização é realizada no objeto de dicionário

## Sintaxe

```
Obj = dict.keys()
```

## Valor de retorno

O método `keys()` retorna o objeto `dict_keys` que é uma visão das chaves no dicionário.

## Exemplo

```
numbers = {10:"Ten", 20:"Twenty", 30:"Thirty",40:"Forty"}
obj = numbers.keys()
print ('type of obj: ', type(obj))
print (obj)
print ("update numbers dictionary")
numbers.update({50:"Fifty"})
print ("View automatically updated")
print (obj)
```

Ele produzirá a seguinte **saída** -

```
type of obj: <class 'dict_keys'>
dict_keys([10, 20, 30, 40])
update numbers dictionary
View automatically updated
dict_keys([10, 20, 30, 40, 50])
```

## Método valores()

O método `values()` retorna uma visão de todos os valores presentes no dicionário. O objeto é do tipo `dict_value`, que é atualizado automaticamente.

## Sintaxe

```
Obj = dict.values()
```

## Valor de retorno

O método `values()` retorna uma visão `dict_values` de todos os valores presentes no dicionário.

## Exemplo

```
numbers = {10:"Ten", 20:"Twenty", 30:"Thirty",40:"Forty"}
obj = numbers.values()
print ('type of obj: ', type(obj))
print (obj)
print ("update numbers dictionary")
numbers.update({50:"Fifty"})
print ("View automatically updated")
print (obj)
```

Ele produzirá a seguinte **saída** -

```
type of obj: <class 'dict_values'>
dict_values(['Ten', 'Twenty', 'Thirty', 'Forty'])
update numbers dictionary
View automatically updated
dict_values(['Ten', 'Twenty', 'Thirty', 'Forty', 'Fifty'])
```