



# Dizionari

Python

- un dizionario è una *collezione* di elementi costituiti da coppie
  - *chiave*
  - *valore*
- ad ogni chiave corrisponde il suo valore
- le chiavi e i valori possono essere di qualsiasi tipo di dato (immutabile per le chiavi)
  - stringhe, numeri ...

## *definizione di un dizionario:*

```
<dizionario> = {"<key>": <val>, "<key>": val, ... "<key>": <val>}
```

```
computer = {"monitor": 220, "CPU": 125.5, "mouse": 15}
```

- analogamente alle liste, i dizionari sono ***dinamici*** e ***mutabili***
- differenze rispetto alle liste:
- ***accesso*** agli elementi della collezione tramite chiave(key) e non tramite indice
- i dizionari ***non*** accettano ***duplicati di chiavi***

- l'operatore [*<chiave>*] permette di accedere al valore
  - per *ottenerlo* (errore se la chiave non è presente)
  - per *modificarlo*
  - per *aggiungere* un nuovo elemento
  - per *rimuovere* un elemento

```
computer = {"monitor": 220, "CPU": 125.5, "mouse": 15}

print(computer["CPU"])           #accesso a un elemento:
                                #output 125.5
print(computer["stampante"])     #accesso errato:
                                #KeyError: 'stampante'

computer["mouse"] = 13.5         #modifica del valore
computer["tastiera"] = 25        #aggiunta di un nuovo elemento
                                #(chiave e valore)
del computer["CPU"]              #eliminazione elemento
```

- **get (<chiave>)**
  - accesso al valore di un elemento
- **keys ()**
  - restituisce l'insieme delle chiavi

```
traduzione = {"uomo":"man", "università":"university", "oggi":"today"}
print(traduzione.get("oggi"))      # output: today

chiavi = traduzione.keys()
print(chiavi)                       #output: dict_keys(['uomo', 'università', 'oggi'])

print(type(chiavi))                 #output: <class 'dict_keys'>

listaChiavi = list(chiavi)
print(listaChiavi)                  #output: ['uomo', 'università', 'oggi']
print(type(listaChiavi))            #output: <class 'list'>
```

- **values ()**
  - restituisce l'insieme dei valori
- **items ()**
  - restituisce l'insieme delle coppie chiave, valore
- **clear ()**
  - rimuove tutti gli elementi

```

valori = traduzione.values()
print(valori)           #output: dict_values(['man', 'university', 'today'])
print(type(valori))    #output: <class 'dict_values'>
listaValori = list(valori)
print(listaValori)     #output: ['man', 'university', 'today']
print(type(listaValori)) #output: <class 'list'>

coppie = traduzione.items()
print(coppie)          #dict_items([('uomo', 'man') ... ('oggi', 'today')])
print(type(coppie))   #<class 'dict_items'>
listaCoppie = list(coppie)
print(listaCoppie)    #[('uomo', 'man'), ... ('oggi', 'today')]
print(type(listaCoppie)) #<class 'list'>

```

- i metodi `keys()`, `values()` e `items()` possono essere utilizzati per iterare sugli elementi di un dizionario
- `in` permette di verificare se una chiave è presente nel dizionario

```
for e in traduzione.keys():
```

```
    print(e,end=" ")
```

```
#output: uomo università oggi
```

```
for v in traduzione.values():
```

```
    print(v,end=" ")
```

```
#output: man university today
```

```
for k,v in traduzione.items():
```

```
    print(k,v,end=" | ")
```

```
#output: uomo man | università university | oggi today |
```

```
if "uomo" in traduzione:
```

```
    print("traduzione di uomo:",traduzione.get("uomo"))
```

- **dict** è un metodo costruttore che prende una sequenza di coppie **<chiave> = <valore>**
- **dict** senza parametri crea un dizionario vuoto
- se il parametro passato a dict è un dizionario viene eseguita una copia di questo
  - anche il metodo **copy()** permette di copiare un dizionario

```

giorni = dict(gen = 31, feb =28, mar=31, apr = 30)
print(giorni)           #output: {'gen': 31, 'feb': 28, 'mar': 31, 'apr': 30}
print(type(giorni))    #output: <class 'dict'>
giorniBisestile = dict(giorni)
giorniBisestile['feb'] = 29
print(giorni)          #{'gen': 31, 'feb': 28, 'mar': 31, 'apr': 30}
print(giorniBisestile) #{'gen': 31, 'feb': 29, 'mar': 31, 'apr': 30}

```



Method	Description
<u>clear</u> ()	Removes all the elements from the dictionary
<u>copy</u> ()	Returns a copy of the dictionary
<u>fromkeys</u> ()	Returns a dictionary with the specified keys and value
<u>get</u> ()	Returns the value of the specified key
<u>items</u> ()	Returns a list containing a tuple for each key value pair
<u>keys</u> ()	Returns a list containing the dictionary's keys
<u>pop</u> ()	Removes the element with the specified key
<u>popitem</u> ()	Removes the last inserted key-value pair
<u>setdefault</u> ()	Returns the value of the specified key. If the key does not exist: insert the key, with the specified value
<u>update</u> ()	Updates the dictionary with the specified key-value pairs
<u>values</u> ()	Returns a list of all the values in the dictionary