

```

import numpy as np

# Iniziamo con un array unidimensionale

x = np.random.randint(4,16,6)
x
array([ 5,  4,  9, 13,  5, 15])

# Supponiamo di voler estrarre solo quegli elementi minori di 7. Il
nostro mask sarà:

mask = x < 7

# A questo punto possiamo utilizzare la nostra funzione where. Essa
estrarrà LA POSIZIONE di quegli elementi
# che soddisfano la nostra condizione mask

posizione_elementi = np.where(mask)
posizione_elementi
(array([0, 1, 4], dtype=int64),)

# Passiamo ora ad una matrice

m = np.random.randint(1,200,(7,7))
m
array([[117, 178,  81, 180, 167,  28,  92],
       [ 56,  21, 165, 141,  34, 159,  93],
       [ 85, 174,  63, 157, 157, 116,   2],
       [178,  29, 122, 152,  88, 164, 160],
       [ 28, 159, 111,  16, 177,  14, 110],
       [ 41, 105, 182, 143,  94,  22,  65],
       [ 80, 149,   1,  52, 165, 119, 127]])

# La funzione where ritornerà un array composto da due sottoarray. Il
primo sottoarray conterrà la riga
# in cui si trovano quegli elementi che soddisfano la nostra
condizione.

```

```
# Il secondo sottoarray conterrà la posizione corrispettiva della colonna.
```

```
mask = m>100
```

```
posizioni = np.where(mask)  
posizioni
```

```
(array([0, 0, 0, 0, 1, 1, 1, 2, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 5,  
5,  
5, 6, 6, 6, 6], dtype=int64),  
array([0, 1, 3, 4, 2, 3, 5, 1, 3, 4, 5, 0, 2, 3, 5, 6, 1, 2, 4, 6, 1,  
2,  
3, 1, 4, 5, 6], dtype=int64))
```

```
# Sostituzione elementi di m con altri elementi
```

```
# Possiamo utilizzare la np.where function per sostituire gli elementi di una matrice.
```

```
# Partiamo da un caso banalissimo. Vogliamo sostituire con A gli elementi > 100 e gli elementi < 155 e con B tutti gli altri.
```

```
# Otterremo:
```

```
mask = (m > 100) && (m < 155)
```

```
np.where(mask, "A", "B")
```

```
array([[ 'A', 'A', 'B', 'A', 'A', 'B', 'B'],  
      [ 'B', 'B', 'A', 'A', 'B', 'A', 'B'],  
      [ 'B', 'A', 'B', 'A', 'A', 'A', 'B'],  
      [ 'A', 'B', 'A', 'A', 'B', 'A', 'A'],  
      [ 'B', 'A', 'A', 'B', 'A', 'B', 'A'],  
      [ 'B', 'A', 'A', 'A', 'B', 'B', 'B'],  
      [ 'B', 'A', 'B', 'B', 'A', 'A', 'A']], dtype='<U1')
```

```
# Consideriamo ora due array di stessa lunghezza. chiamiamo ora un array studenti e uno voti. I voti minori di 18 faranno
```

```
# Bocciare gli studenti, mentre voti > li faranno promuovere
```

```
# Consideriamo 15 studenti e i rispettivi voti
```

```

studenti =
np.array(["Mario","Luca","Giada","Franca","Gertrudina","Fava","Astolfo",
"","Astoide","Gervasio","Kara","Bòe","Gina","Gino","Uga","Fabiola"])
studenti

array(['Mario', 'Luca', 'Giada', 'Franca', 'Gertrudina', 'Fava',
      'Astolfo', 'Astoide', 'Gervasio', 'Kara', 'Bòe', 'Gina',
      'Gino',
      'Uga', 'Fabiola'], dtype='<U10')

voti = np.random.randint(0,30,15)
voti

array([23, 20, 23, 14, 26, 27, 15, 10, 11, 25, 14, 15,  0, 20, 25])

mask = (voti>=18)

alunni_promossi = np.where(mask,"BOCCIATO",studenti)
alunni_promossi

array(['BOCCIATO', 'BOCCIATO', 'BOCCIATO', 'Franca', 'BOCCIATO',
      'BOCCIATO', 'Astolfo', 'Astoide', 'Gervasio', 'BOCCIATO',
      'Bòe',
      'Gina', 'Gino', 'BOCCIATO', 'BOCCIATO'], dtype='<U10')

```

