

Hướng dẫn sử dụng Structure Array trong NumPy

Giáo viên: Nguyễn Hùng Cường

Trong NumPy có cung cấp một Structure Array tương tự như kiểu dữ liệu cấu trúc (struct) trong C. Mảng cấu trúc được dùng để nhóm các dữ liệu thuộc các kiểu dữ liệu và kích thước khác nhau. Các structure array có chứa các field để chứa dữ liệu. Mỗi trường dữ liệu có thể chứa dữ liệu thuộc các kiểu và kích thước khác nhau. Các phần tử mảng có thể được truy cập thông qua dấu chấm.

Lưu ý:

- Các cấu trúc trong mảng có cùng số lượng các field.
- Các cấu trúc có cùng các tên trường.

Bước1: Khai báo và khởi tạo mảng cấu trúc

Mở một trình soạn thảo, sau đó định nghĩa một kiểu cấu trúc, rồi khai báo một mảng cấu trúc với kiểu cấu trúc vừa định nghĩa ở trên.

```
DemoNumpyStructureArray.py x
1  import numpy as np
2
3  # khai báo kiểu dữ liệu structure
4  seafood_type = [('name', (np.str_, 30)), ('age', np.int32), ('address', np.str_, 50)]
5
6  # khai báo và khởi tạo Structure Array
7  a = np.array([('Tiểu Vy', 23, "Settle"), ('Đăng Thu Thảo', 30, "Miami"),
8               ('Mai Bella', 28, "Zurich"), ('Thúy Vân', 29, "Berlin"),
9               ('Phượng Nhi', 22, "Hawaii"), ('Mai Ngọc', 32, "New York"),
10              ('Hà Kiều Anh', 45, "Texas"), ('Nguyễn Trần Khánh Vân', 24, "Paris")],
11              dtype=seafood_type)
```

Sau đó, ta có thể tiến hành sắp xếp dữ liệu trong mảng bằng cách gọi hàm sort() như sau:

```

13 # Sắp xếp mảng theo thuộc tính name
14 b = np.sort(a, order='name')
15 print('Sắp xếp mảng theo thuộc tính name', b)
16
17 # Sắp xếp mảng theo thuộc tính age
18 b = np.sort(a, order='age')
19 print('\nSắp xếp mảng theo thuộc tính age', b)
20
21 # sắp xếp mảng theo trường age theo thứ tự giảm dần
22 sorted_data = np.sort(a, order='age')[::-1]
23 print('Sắp xếp mảng theo thứ tự giảm dần: ', sorted_data)

```

Tiếp đó, ta thực thi chương trình thì sẽ thấy mảng đã được sắp xếp dữ liệu đúng như mong muốn.

```

DemoNumpyStructureArray x
"C:\Program Files\Python 3.11\python.exe" G:/Vidubaigiangdemo/DemoPython2022/DemoNumPy/DemoNumpyStructureArray.py
Sắp xếp mảng theo thuộc tính name [('Hà Kiều Anh', 45, 'Texas') ('Mai Bella', 28, 'Zurich')
('Mai Ngọc', 32, 'New York') ('Nguyễn Trần Khánh Vân', 24, 'Paris')
('Phương Nhi', 22, 'Hawaii') ('Thúy Vân', 29, 'Berlin')
('Tiểu Vy', 23, 'Settle') ('Đặng Thu Thảo', 30, 'Miami')]

Sắp xếp mảng theo thuộc tính age [('Phương Nhi', 22, 'Hawaii') ('Tiểu Vy', 23, 'Settle')
('Nguyễn Trần Khánh Vân', 24, 'Paris') ('Mai Bella', 28, 'Zurich')
('Thúy Vân', 29, 'Berlin') ('Đặng Thu Thảo', 30, 'Miami')
('Mai Ngọc', 32, 'New York') ('Hà Kiều Anh', 45, 'Texas')]
Sắp xếp mảng theo thứ tự giảm dần: [('Hà Kiều Anh', 45, 'Texas') ('Mai Ngọc', 32, 'New York')
('Đặng Thu Thảo', 30, 'Miami') ('Thúy Vân', 29, 'Berlin')
('Mai Bella', 28, 'Zurich') ('Nguyễn Trần Khánh Vân', 24, 'Paris')
('Tiểu Vy', 23, 'Settle') ('Phương Nhi', 22, 'Hawaii')]

```

Tiếp theo, ta có thể lấy về các giá trị min, max, mean trong mảng với các hàm tương ứng như sau.

Bên cạnh đó, NumPy còn cung cấp cơ chế filter dữ liệu trong mảng. Ở bên dưới ta đã lọc ra những phần tử seafood với age <= 30 rồi hiển thị ra như hình bên dưới.

```

25     # lay ve gia tri min, max, mean trong mang
26     averageage = np.mean(a['age'])
27     maxage = np.max(a['age'])
28     minage = np.min(a['age'])
29     print('Tuoi trung binh: ', averageage)
30     print('Tuoi lon nhat: ', maxage)
31     print('Tuoi nho nhat: ', minage)
32
33     # filter du lieu trong structure array
34     filtered_data = a[a['age'] <= 30]
35     print(filtered_data)

```

Sau đó, thực thi ứng dụng ta có thể thấy dữ liệu đã được hiển thị ra đúng như mong muốn.

```

Tuoi trung binh:  29.125
Tuoi lon nhat:   45
Tuoi nho nhat:  22
[('Tiểu Vy', 23, 'Settle') ('Đặng Thu Thảo', 30, 'Miami')
 ('Mai Bella', 28, 'Zurich') ('Thuý Vân', 29, 'Berlin')
 ('Phương Nhi', 22, 'Hawaii') ('Nguyễn Trần Khánh Vân', 24, 'Paris')]

```

Ngoài ra, NumPy cũng hỗ trợ thao tác nối mảng, và reshape mảng như sau. Ở đây ta đã nối thêm một mảng gồm 2 thông tin nhân viên, sau đó ta đã reshape mảng này thành một mảng gồm 4 dòng/2 cột như hình bên dưới.

```

38 # concatenation structure array
39 new_seafood = np.array([('Trương Ngọc Ánh', 44, "Madrid"), ('Đỗ Thị Hà', 23, "Barcelonna")], dtype=seafood_type)
40 concatenated_data = np.concatenate((a, new_seafood))
41 print('Mang sau khi được concat là: ', concatenated_data)
42
43 # thực hiện reshape structure array
44 reshaped_data = np.reshape(a, (4, 2))
45 print('Mang sau khi được reshape là: ', reshaped_data)

```

Sau khi đã viết mã xong, hãy thực thi ứng dụng, ta sẽ thấy mảng sau khi được nối và reshape đã hiển thị kết quả đúng như mong muốn.

```

Mang sau khi được concat là: [('Tiểu Vy', 23, 'Settle') ('Đặng Thu Thảo', 30, 'Miami')
('Mai Bella', 28, 'Zurich') ('Thúy Vân', 29, 'Berlin')
('Phương Nhi', 22, 'Hawaii') ('Mai Ngọc', 32, 'New York')
('Hà Kiều Anh', 45, 'Texas') ('Nguyễn Trần Khánh Vân', 24, 'Paris')
('Trương Ngọc Ánh', 44, 'Madrid') ('Đỗ Thị Hà', 23, 'Barcelonna')]
Mang sau khi được reshape là: [[('Tiểu Vy', 23, 'Settle') ('Đặng Thu Thảo', 30, 'Miami')]
[('Mai Bella', 28, 'Zurich') ('Thúy Vân', 29, 'Berlin')]
[('Phương Nhi', 22, 'Hawaii') ('Mai Ngọc', 32, 'New York')]
[('Hà Kiều Anh', 45, 'Texas') ('Nguyễn Trần Khánh Vân', 24, 'Paris')]]

```