FREQUENTLY ASKED QUESTIONS:

PYTHON: NUMPY

PART 2

FAQ

1)How to replace items that satisfy a condition without affecting the original array?

```
import numpy as np

arr = np.arange(10)
out = np.where(arr % 2 == 1, -1, arr)
print(arr)
print(out)
```

Output:

```
[0 1 2 3 4 5 6 7 8 9]
[ 0 -1  2 -1  4 -1  6 -1  8 -1]
```

2) How to stack two arrays vertically?

```
import numpy as np

a = np.arange(10).reshape(2,-1)
b = np.repeat(1, 10).reshape(2,-1)

# Answers
# Method 1:
np.concatenate([a, b], axis=0)


# Method 2:
np.vstack([a, b])

# Method 3:
np.r_[a, b]
```

Output:

```
#> array([[0, 1, 2, 3, 4],
#>        [5, 6, 7, 8, 9],
#>        [1, 1, 1, 1, 1],
#>        [1, 1, 1, 1, 1]])
```

FAQ

3) How to print only 3 decimal places in python numpy array?

```
import numpy as np

# Input
rand_arr = np.random.random((5,3))

# Create the random array
rand_arr = np.random.random([5,3])

# Limit to 3 decimal places
np.set_printoptions(precision=3)
rand_arr[:4]
```

Output:

```
[[0.998 0.071 0.591]
 [0.835 0.502 0.452]
 [0.921 0.995 0.672]
 [0.994 0.306 0.668]
 [0.282 0.422 0.778]]
```

4)How to limit the number of items printed in output of numpy array?

```
import numpy as np

np.set_printoptions(threshold=6)
a=np.arange(15)
print(a)
```

Output:

```
[ 0  1  2 ... 12 13 14]
```

FAQ

5)How to print the full numpy array without truncating?

```
import numpy as np
np.set_printoptions(threshold=6)
a = np.arange(15)

# Solution
np.set_printoptions(threshold=np.nan)
print(a)
```

6)How to import a dataset with numbers and texts keeping the text intact in python numpy?

```
import numpy as np
# Solution
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'
iris = np.genfromtxt(url, delimiter=',', dtype='object')
names = ('sepallength', 'sepalwidth', 'petallength', 'petalwidth', 'species')

# Print the first 3 rows
print(iris[:3])
```

Output:

```
[[b'5.1' b'3.5' b'1.4' b'0.2' b'Iris-setosa']
 [b'4.9' b'3.0' b'1.4' b'0.2' b'Iris-setosa']
 [b'4.7' b'3.2' b'1.3' b'0.2' b'Iris-setosa']]
```

7)How to compute the min-by-max for each row for a numpy array 2d?

```
import numpy as np
np.random.seed(100)
a = np.random.randint(1,10, [5,3])

print(a)

# Solution
print(np.apply_along_axis(lambda x: np.min(x)/np.max(x), arr=a, axis=1))
```

FAQ

8)How to find the duplicate records in a numpy array?

```
import numpy as np
# Input
np.random.seed(100)
a = np.random.randint(0, 5, 10)

## Solution

# Create an all True array
out = np.full(a.shape[0], True)

# Find the index positions of unique elements
unique_positions = np.unique(a, return_index=True)[1]

# Mark those positions as False
out[unique_positions] = False
print(out)
```

Output:

```
[False  True False ...  True  True  True]
```

9) How to drop all missing values from a numpy array?

```
import numpy as np
a = np.array([1,2,3,np.nan,5,6,7,np.nan])
a[~np.isnan(a)]
print(a)
```

FAQ

10) How to find the index of n'th repetition of an item in an array?

```
import numpy as np
x = np.array([1, 2, 1, 1, 3, 4, 3, 1, 1, 2, 1, 1, 2])
n = 5

# Solution 1: List comprehension
[i for i, v in enumerate(x) if v == 1][n-1]

# Solution 2: Numpy version
np.where(x == 1)[0][n-1]
```

11) How to subtract a 1d array from a 2d array, where each item of 1d array subtracts from respective row?

```
import numpy as np
# Input
a_2d = np.array([[3,3,3],[4,4,4],[5,5,5]])
b_1d = np.array([1,2,3])

# Solution
print(a_2d - b_1d[:,None])
```

Output:

```
[[2 2 2]
 [2 2 2]
 [2 2 2]]
```

FAQ

12) How to convert an array of arrays into a flat 1d array?

```
import numpy as np
# Input
arr1 = np.arange(3)
arr2 = np.arange(3,7)
arr3 = np.arange(7,10)

array_of_arrays = np.array([arr1, arr2, arr3])
print('array_of_arrays: ', array_of_arrays)

# Solution 1
arr_2d = np.array([a for arr in array_of_arrays for a in arr])

# Solution 2:
arr_2d = np.concatenate(array_of_arrays)
print(arr_2d)
```

Output:

```
array_of_arrays:  [array([0, 1, 2]) array([3, 4, 5, 6]) array([7, 8, 9])]
[0 1 2 ... 7 8 9]
```

13) How to replace all values greater than a given value to a given cutoff?

```
import numpy as np
# Input
np.set_printoptions(precision=2)
np.random.seed(100)
a = np.random.uniform(1,50, 20)

# Solution 1: Using np.clip
np.clip(a, a_min=10, a_max=30)

# Solution 2: Using np.where
print(np.where(a < 10, 10, np.where(a > 30, 30, a)))
```

Output:

```
[27.63 14.64 21.8  ... 10.   30.   14.43]
```

FAQ

14) How to get the second largest value of an array when grouped by another array?

```
import numpy as np
# Import iris keeping the text column intact
url = 'https://archive.ics.uci.edu/ml/machine-learning-databases/iris/iris.data'
iris = np.genfromtxt(url, delimiter=',', dtype='object')

# Solution
# Get the species and petal length columns
petal_len_setosa = iris[iris[:, 4] == b'Iris-setosa', [2]].astype('float')

# Get the second last value
np.unique(np.sort(petal_len_setosa))[-2]
```

15)How to rank items in an array using numpy?

```
import numpy as np
np.random.seed(10)
a = np.random.randint(20, size=10)
print('Array: ', a)

# Solution
print(a.argsort().argsort())
print('Array: ', a)
```

Output:

```
Array:  [ 9  4 15 ...  8  9  0]
[4 2 6 ... 3 5 1]
Array:  [ 9  4 15 ...  8  9  0]
```

FAQ

16) How to rank items in a multidimensional array using numpy?

```
import numpy as np
# Input:
np.random.seed(10)
a = np.random.randint(20, size=[2,5])
print(a)

# Solution
print(a.ravel().argsort().argsort().reshape(a.shape))
```

Output:

17)How to convert numpy's datetime64 object to datetime's datetime object?

```
import numpy as np
# Input: a numpy datetime64 object
dt64 = np.datetime64('2018-02-25 22:10:10')

# Solution
from datetime import datetime
dt64.tolist()

# or

dt64.astype(datetime)
#> datetime.datetime(2018, 2, 25, 22, 10, 10)
```

18)How to create a numpy array sequence given only the starting point, length and the step?

```
import numpy as np
length = 10
start = 5
step = 3

def seq(start, length, step):
    end = start + (step*length)
    return np.arange(start, end, step)

seq(start, length, step)
```

FAQ

19) Create a 4X2 integer array and Prints its attributes

```
import numpy as np
firstArray = np.empty([4,2], dtype = np.uint16)
print("Printing Array")
print(firstArray)

print("Printing numpy array Attributes")
print("1> Array Shape is: ", firstArray.shape)
print("2>. Array dimensions are ", firstArray.ndim)
print("3>. Length of each element of array in bytes is ", firstArray.itemsize)
```

Output:

```
Printing Array
[[22512 56396]
 [  603     0]
 [    0     0]
 [    0     0]]
Printing numpy array Attributes
1> Array Shape is:  (4, 2)
2>. Array dimensions are  2
3>. Length of each element of array in bytes is  2
```

20) How to print the full numpy array without truncating?

```
import numpy as np
np.set_printoptions(threshold=6)
a = np.arange(15)

# Solution
np.set_printoptions(threshold=np.nan)
print(a)
```

Output:

```
([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14])
```