

```
In [20]: #Task 1:
#Stack arrays a and b vertically
import numpy as np

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

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

x = np.vstack((a,b))
print('output = ')
print (x)
```

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

```
In [21]: #Task 2
#Create the following pattern without hardcoding. Use only numpy functions
#and the below input array a.
```

```
a = np.array([1,2,3])

#print(a)

x= np.repeat(a,3)
#print (x)

x= np.concatenate((x,a,a,a))
print(x)
```

```
[1 1 1 2 2 2 3 3 3 1 2 3 1 2 3 1 2 3]
```

In [22]: *#Task 3*

#Convert the function maxx that works on two scalars, to work on two arrays.

```
a = np.array([5, 7, 9, 8, 6, 4, 5])
```

```
b = np.array([6, 3, 4, 8, 9, 7, 1])
```

```
x= np.max((a,b),axis=0)
```

```
print (x)
```

```
[6 7 9 8 9 7 5]
```

In [23]: *#Task 4*

#Swap rows 1 and 2 in the array arr:

```
v = np.vstack((b,a))
```

```
print(v)
```

```
[[6 3 4 8 9 7 1]
```

```
 [5 7 9 8 6 4 5]]
```

```
In [24]: #Task 5
#From array a remove all items present in array b
#Input:

import numpy as np

a = np.array([1, 2, 3, 4, 5])
b = np.array([5, 6, 7, 8, 9])

result = np.setdiff1d(a, b) # return unique value that are not in array b
print("Result:", result)
```

Result: [1 2 3 4]

In []: