

ASSIGNMENT 1

1.Import numpy as np and print the version number.

```
In [1]: import numpy as np
```

```
In [3]: print(np.__version__)
```

```
1.24.4
```

2. Create a 1D array of numbers from 0 to 9

```
In [2]: arr = np.arange(10)
print(arr)
```

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

3. Create a 3×3 numpy array of all True's.

```
In [4]: arr = np.full((3, 3), True)
print(arr)
```

```
[[ True  True  True]
 [ True  True  True]
 [ True  True  True]]
```

4.Extract all odd numbers from arr

```
In [5]: arr = np.arange(10)
odd_numbers = arr[arr % 2 != 0]
print(odd_numbers)
```

```
[1 3 5 7 9]
```

5. Replace all odd numbers in arr with -1

```
In [6]: arr = np.arange(10)
arr[arr % 2 != 0] = -1
print(arr)
```

```
[ 0 -1  2 -1  4 -1  6 -1  8 -1]
```

6.Replace all odd numbers in arr with -1 without changing arr

```
In [7]: arr = np.arange(10)
new_arr = np.where(arr % 2 != 0, -1, arr)
print(new_arr)
print(arr)
```

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

7.Convert a 1D array to a 2D array with 2 rows

```
In [9]: arr = np.arange(10)
new_arr = arr.reshape(2, 5)
print(new_arr)
```

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

8.Stack arrays a and b vertically

```
In [10]: a = np.array([1, 2, 3])
b = np.array([4, 5, 6])
arr5 = np.vstack((a, b))
print(arr5)
```

```
[[1 2 3]
 [4 5 6]]
```

9. Stack the arrays a and b horizontally.

```
In [11]: arr6 = np.hstack((a, b))
print(arr6)
```

```
[1 2 3 4 5 6]
```

10.Create the following pattern without hardcoding. Use only numpy functions and the below input array a**

```
In [13]: a = np.array([1, 2, 3, 4, 5, 6])
arr7 = a.repeat(a)
print(arr7)
```

```
[1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 6 6 6 6 6 6]
```

11.Get the common items between a and b

```
In [14]: a = np.array([1, 2, 3, 4, 5])
b = np.array([4, 5, 6, 7, 8])
common_items = np.intersect1d(a, b)
print(common_items)
```

```
[4 5]
```

12. From array a remove all items present in array b

```
In [15]: diff = np.setdiff1d(a, b)
         print(diff)
```

```
[1 2 3]
```

13. Get the positions where elements of a and b match

```
In [18]: match_positions = np.where(a == b)[0]
         print(match_positions)
```

```
[]
```

14. Get all items between 5 and 10 from a.

```
In [19]: a = np.array([1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12])
         items_between_5_and_10 = a[(a >= 5) & (a <= 10)]
         print(items_between_5_and_10)
```

```
[ 5  6  7  8  9 10]
```

15. Convert the function max that works on two scalars, to work on two arrays.

```
In [20]: def maxx(x, y):
         return np.maximum(x, y)
```

```
a = np.array([1, 2, 3, 4])
b = np.array([5, 6, 7, 8])
result = maxx(a, b)
print(result)
```

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

16. Swap columns 1 and 2 in the array arr.

```
In [21]: arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])
         arr[:, [1, 0, 2]] = arr[:, [0, 1, 2]]
         print(arr)
```

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

17. Swap rows 1 and 2 in the array arr:

```
In [22]: arr[[1, 0, 2]] = arr[[0, 1, 2]]  
print(arr)
```

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

18. Reverse the rows of a 2D array arr

```
In [23]: arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])  
reversed_rows = arr[::-1]  
print(reversed_rows)
```

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

19. Reverse the columns of a 2D array arr.

```
In [25]: arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])  
reversed_columns = arr[:, ::-1]  
print(reversed_columns)
```

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

20. Create a 2D array of shape 5x3 to contain random decimal numbers between 5 and 10.

```
In [26]: arr = np.random.uniform(5, 10, size=(5, 3))  
print(arr)
```

```
[[6.65419426 7.51162419 6.27008296]  
 [6.13743722 6.02399631 6.89323813]  
 [7.26421631 7.19234284 8.14475119]  
 [8.753817   5.78849431 7.83578898]  
 [6.11624949 5.53295676 5.16869716]]
```

```
In [ ]:
```