
output

Q.NO 1

```
main.py x
1 import numpy as np
2 S = np.array([8,9,10,11,12])
3 nz = 5
4 P = np.zeros(len(S) + (len(S)-1)*(nz))
5 P[::nz+1] = S
6 print(P)
```

```
Console Shell
[ 8.  0.  0.  0.  0.  0.  9.  0.  0.  0.  0.  0. 10.  0.  0.  0
.  0.  0.
11.  0.  0.  0.  0.  0. 12.]
>
```

Q.NO 2

```
main.py x
1 import numpy as np
2 a = np.random.randint(0,2,5)
3 print("First array:")
4 print(a)
5 b = np.random.randint(0,2,5)
6 print("Second array:")
7 print(b)
8 print("Testing above two arrays are same or not!")
9 array_equal = np.allclose(a, b)
10 print(array_equal)
```

```
Console Shell
First array:
[1 0 1 0 1]
Second array:
[1 0 0 1 1]
Testing above two arrays are same or not!
False
>
```

Q.NO 3

```
main.py ×  
1 import numpy as np  
2 print(0 * np.nan)  
3 print(np.nan != np.nan)  
4 print(np.inf > np.nan)  
5 print(np.nan - np.nan)  
6 print(0.3 == 3 * 0.1)  
7
```

Console Shell

```
nan  
True  
False  
nan  
False  
>
```

Q.NO 4

```
main.py ×  
1 import pandas as pd  
2 ser = pd.Series(['amrita', 'school', 'of',  
3 'engineering', 'chennai', 'campus'])  
4 Series = ser.str.title()  
5 print(Series)  
6
```

Console Shell

```
0      Amrita  
1      School  
2      Of  
3  Engineering  
4      Chennai  
5      Campus  
dtype: object  
>
```

Q.NO 5-1

```
main.py ×  
1  
2 import numpy as np  
3  
4 a = np.array([2, 2, 2,2,2])  
5 b = np.array([6, 6, 6,6,6])  
6  
7 print ("Array 1 : ", a)  
8 print ("\nArray 2 : ", b)  
9  
10 out_ab = np.add(a, b)  
11 print ("\nFinal : ", out_ab)
```

Console Shell

```
Array 1 : [2 2 2 2 2]  
  
Array 2 : [6 6 6 6 6]  
  
Final : [8 8 8 8 8]  
➤
```

Q.NO 5-3

```
ain.py ×  
1 import numpy as sai  
2 b = sai.identity(2, dtype = float)  
3 print("Matrix b : \n", b)  
4 p = sai.identity(4)  
5 print("\nMatrix p: \n", p)  
6
```

Console Shell

```
Matrix b :  
[[1. 0.]  
 [0. 1.]]  
  
Matrix p:  
[[1. 0. 0. 0.]  
 [0. 1. 0. 0.]  
 [0. 0. 1. 0.]  
 [0. 0. 0. 1.]]  
➤
```