



## Tool & Technique Laboratory (T&T Lab.)

[CS-3096]

### Individual Work

**Lab. No:7 , Date:28-02-2023 , Day:Tuesday**

### Topic:

Roll Number:	20051939	Branch/Section:	CSE-17
Name in Capital:	Shashikant shah		

**(Instruction:** Rename this file as r-LAB-x where r is your roll number & x is your lab. number & Suppose your roll number is 1905123 & you want to submit lab-2 programs, then file name should be 1905123-LAB-2. Finally delete all texts inside parentheses, also parenthesis)

### Program No: (1)

#### Program Title:

()# 1. Write a NumPy program to concatenate element-wise two arrays of string.

# Sample Output:

# Array1:

# ['Python' 'PHP' ]

# Array2:

# [' Java' ' C++']

# new array:

# ['Python Java' 'PHP C++']

### Input/Output Screenshots:

#### RUN-1:

(Paste here the screenshots of first run)

```
Enter the number of elements in the array: 2
Enter the elements of the array:
11
22
Enter the elements of the array:
33
44
New array: { 11 33 22 44
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniqu
```

#### RUN-2

(Paste here the screenshots of second run )

```

Techniques Laboratory\lab 7\lq.py
Enter the number of elements in the array: 2
Enter the elements of the array:
aa
bb
Enter the elements of the array:
aa
ccc
New array: { aa aa }bb ccc
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques Labora

```

### Source code

(Paste here the source code)

```

n = int(input("Enter the number of elements in the array: "))
print("Enter the elements of the array: ")
arr1 = np.array([input() for i in range(n)])
print("Enter the elements of the array: ")
arr2 = np.array([input() for i in range(n)])
print("New array: {", end = " ")
# add space between the elements of the array
new_arr = []
for i in range(n):
    new_arr.append(arr1[i] + " " + arr2[i])
print(" ".join(new_arr), end = " ")
#

```

### Conclusion/Observation

successfully compiled the code

**Program No: (2)****Program Title:**

(# 2. Write a NumPy program to split the element of a given array with spaces.

# Sample Output:

# Original Array:

# ['Python PHP Java C++']

# Split the element of the said array with spaces:

# [list(['Python', 'PHP', 'Java', 'C++'])]

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)

```

● Techniques Laboratory\lab 7\2q.py
Original Array:  ['Python PHP julia helloworld']
Split the element of the said array with spaces:  [list(['Python', 'PHP', 'julia', 'helloworld'])]
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques Laboratory\lab 7>
Ln 11, Col 45  Spaces: 4  UTF-8  CRLF  {Python}  3.10.10 64-bit (microsoft store)  Go Live  Spell  Colorize: 0

```

**Source code**

(Paste here the source code)

```

import numpy as np

arr = np.array(['Python PHP julia helloworld'])
print("Original Array: ", arr)
print("Split the element of the said array with spaces: ", np.char.split(arr))

```

**Conclusion/Observation**

successfully compiled the code

**Program No: (3)****Program Title:**

(# 3. Write a NumPy program to count the lowest index of "P" in a given array, element-wise.

# Sample Output:

# Original Array:

# ['Python' 'PHP' 'JS' 'EXAMPLES' 'HTML']

# count the lowest index of 'P':

# [ 0 0 -1 4 -1]

)

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)

```
Techniques Laboratory\lab 7\3q.py"
Original Array: ['Python' 'PHP' 'JS' 'java' 'react js']
count the lowest index of 'P': [ 0  0 -1 -1 -1]
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques Laboratory\lab 7>
```

**Source code**

(Paste here the source code)

```
import numpy as np
arr = np.array(['Python', 'PHP', 'JS', 'java', 'react js'])
print("Original Array: ", arr)
print("count the lowest index of 'P': ", np.char.find(arr, 'P'))
```

**Conclusion/Observation**

successfully compiled the code

**Program No: (4)****Program Title:**

(# 4. Write a NumPy program to count a given word in each row of a given array of string values.

Sample output:

# Original array of string values:

# [['Python' 'NumPy' 'Exercises']

# ['Python' 'Pandas' 'Exercises']

# ['Python' 'Machine learning' 'Python']]

# Count 'Python' row wise in the above array of string values:

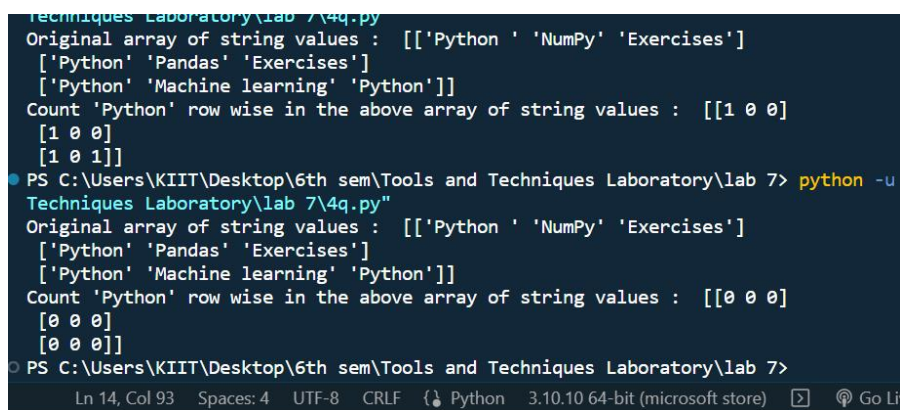
# [[1 0 0]

# [1 0 0]

# [1 0 1]]]

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)



```
Techniques Laboratory\lab 7\4q.py
Original array of string values : [['Python ' 'NumPy' 'Exercises']
['Python' 'Pandas' 'Exercises']
['Python' 'Machine learning' 'Python']]
Count 'Python' row wise in the above array of string values : [[1 0 0]
[1 0 0]
[1 0 1]]
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques Laboratory\lab 7> python -u
Techniques Laboratory\lab 7\4q.py
Original array of string values : [['Python ' 'NumPy' 'Exercises']
['Python' 'Pandas' 'Exercises']
['Python' 'Machine learning' 'Python']]
Count 'Python' row wise in the above array of string values : [[0 0 0]
[0 0 0]
[0 0 0]]
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques Laboratory\lab 7>
```

**Source code**

(Paste here the source code)

```
import numpy as np
arr = np.array(['Python ', 'NumPy', 'Exercises'], ['Python', 'Pandas', 'Exercises'], ['Python', 'Machine learning', 'Python'])
print("Original array of string values : ", arr)
print("Count 'Python' row wise in the above array of string values : ", np.char.count(arr, 'Python'))
```

**Conclusion/Observation**

successfully compiled the code

**Program No: (5)****Program Title:**

(# 5. WAP to print max from axis 0 and min from axis 1 from the following 2-D array.

# Sample output:

# Printing Original array

# [[34 43 73]

# [82 22 12]

# [53 94 66]]

# Printing amin Of Axis 1

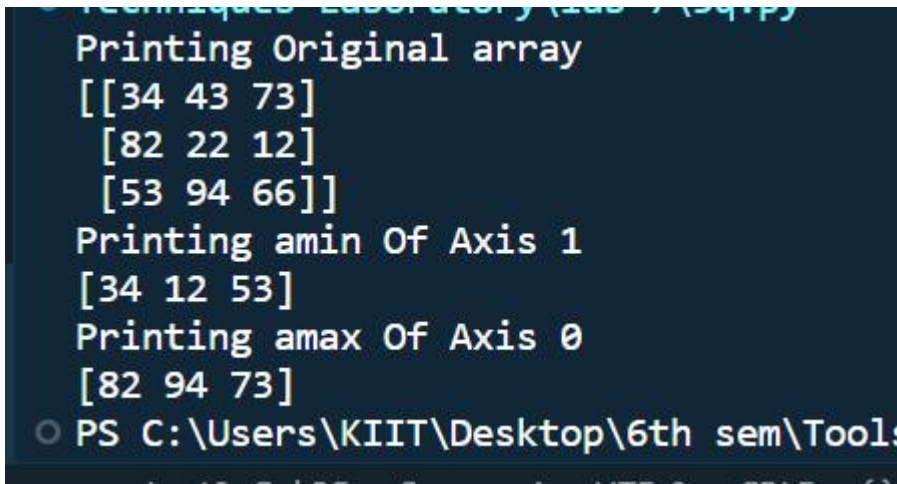
# [34 12 53]

# Printing amax Of Axis 0

# [82 94 73])

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)



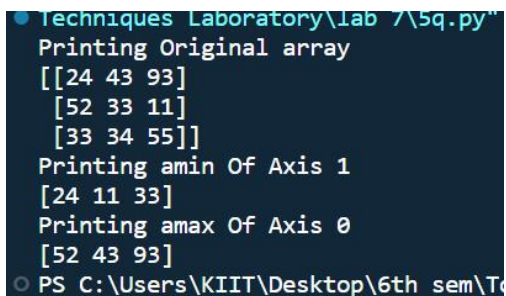
```

Techniques Laboratory\lab 7\5q.py
Printing Original array
[[34 43 73]
 [82 22 12]
 [53 94 66]]
Printing amin Of Axis 1
[34 12 53]
Printing amax Of Axis 0
[82 94 73]
PS C:\Users\KIIT\Desktop\6th sem\Tools

```

**RUN-2**

(Paste here the screenshots of second run )



```

Techniques Laboratory\lab 7\5q.py
Printing Original array
[[24 43 93]
 [52 33 11]
 [33 34 55]]
Printing amin Of Axis 1
[24 11 33]
Printing amax Of Axis 0
[52 43 93]
PS C:\Users\KIIT\Desktop\6th sem\Tools

```

**Source code**

(Paste here the source code)

```
import numpy as np
arr = np.array([[24, 43, 93], [52, 33, 11], [33, 34, 55]])
print("Printing Original array")
print(arr)
print("Printing amin Of Axis 1")
print(np.amin(arr, axis = 1)) # axis = 1 means row wise operation and amin use to find minimum value
print("Printing amax Of Axis 0") # axis = 0 means column wise operation and amax use to find maximum value
print(np.amax(arr, axis = 0))
```

**Conclusion/Observation**

successfully compiled the code

**Program No: (6)****Program Title:**

(# 6. WAP to delete the second column from a given array and insert the following new column in its place.

# Sample output:

# Printing Original array

# [[34 43 73]

# [82 22 12]

# [53 94 66]]

# Array after deleting column 2 on axis 1

# [[34 73]

# [82 12]

# [53 66]]

# Array after inserting column 2 on axis 1

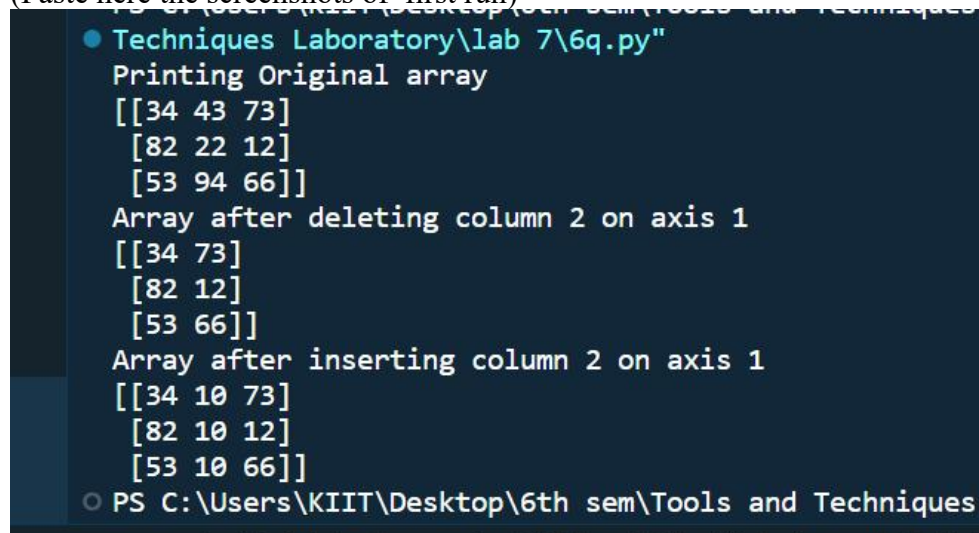
# [[34 10 73]

# [82 10 12]

# [53 10 66]])

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)



```

● Techniques Laboratory\lab 7\6q.py"
Printing Original array
[[34 43 73]
 [82 22 12]
 [53 94 66]]
Array after deleting column 2 on axis 1
[[34 73]
 [82 12]
 [53 66]]
Array after inserting column 2 on axis 1
[[34 10 73]
 [82 10 12]
 [53 10 66]]
○ PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques

```

**RUN-2**

(Paste here the screenshots of second run )



```

• Techniques Laboratory\lab 7\6q.py"
Printing Original array
[[44 53 83]
 [72 33 23]
 [43 54 44]]
Array after deleting column 2 on axis 1
[[44 83]
 [72 23]
 [43 44]]
Array after inserting column 2 on axis 1
[[44 10 83]
 [72 10 23]
 [43 10 44]]
○ PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniq

```

### Source code

(Paste here the source code)

```

import numpy as np
arr = np.array([[44, 53, 83], [72, 33, 23], [43, 54, 44]])
print("Printing Original array")
print(arr)
print("Array after deleting column 2 on axis 1")
arr = np.delete(arr, 1, axis = 1)
print(arr)
print("Array after inserting column 2 on axis 1")
arr = np.insert(arr, 1, 10, axis = 1)
print(arr)

```

### Conclusion/Observation

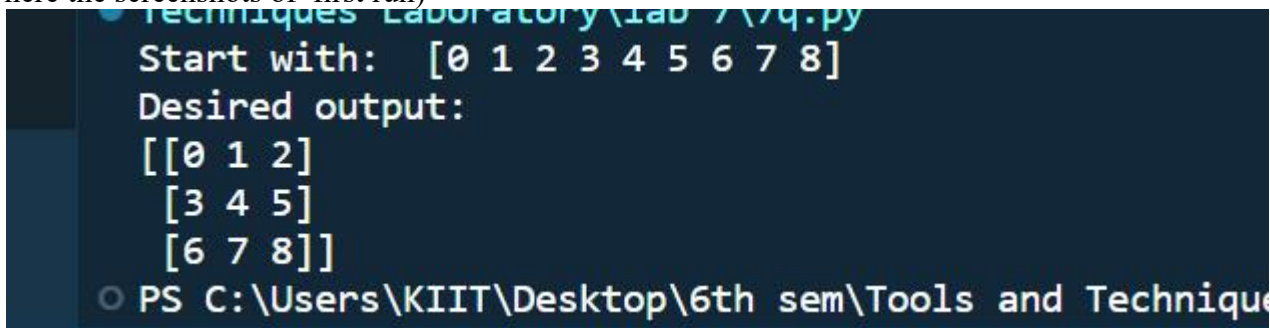
successfully compiled the code

**Program No: (7)****Program Title:**

(Write here your program title in detail)

**# 7. WAP to Convert a 1-D array into a 2-D array with 3 rows****# Sample output:****# Start with: exercise 2 = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8])****# Desired output:****# [[ 0, 1, 2]****# [3, 4, 5]****# [6, 7, 8]]****Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)



```

Techniques Laboratory\lab 7\7q.py
Start with: [0 1 2 3 4 5 6 7 8]
Desired output:
[[0 1 2]
 [3 4 5]
 [6 7 8]]
PS C:\Users\KIIT\Desktop\6th sem\Tools and Techniques>

```

**Source code**

(Paste here the source code)

```

import numpy as np
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8])
print("Start with: ", arr)
print("Desired output: ")
print(arr.reshape(3, 3))

```

**Conclusion/Observation**

successfully compiled the code

**Program No: (8)****Program Title:**

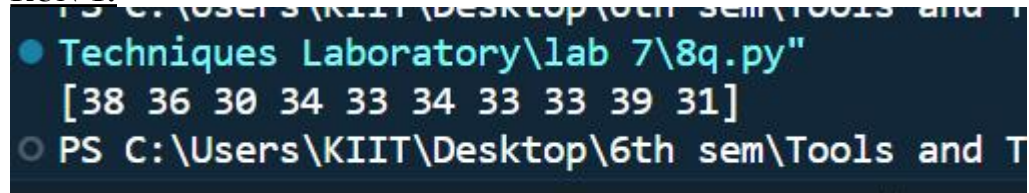
(# 8. WAP to generate a 1-D array of 10 random integers. Each integer should be a number between 30 and 40 (inclusive)

# Sample output:

# [36, 30, 36, 38, 31, 35, 36, 30, 32, 34])

**Input/Output Screenshots:**

**RUN-1:**



```
PS C:\Users\KIIT\Desktop\6th sem\Tools and T
● Techniques Laboratory\lab 7\8q.py"
  [38 36 30 34 33 34 33 33 39 31]
○ PS C:\Users\KIIT\Desktop\6th sem\Tools and T
```

**Source code**

(Paste here the source code)

```
import numpy as np
arr = np.random.randint(30, 41, 10)
print(arr)
```

**Conclusion/Observation**

successfully compiled the code

**Program No: (9)****Program Title:**

# 9. WAP to Replace all odd numbers in the given array with -1

# Sample output:

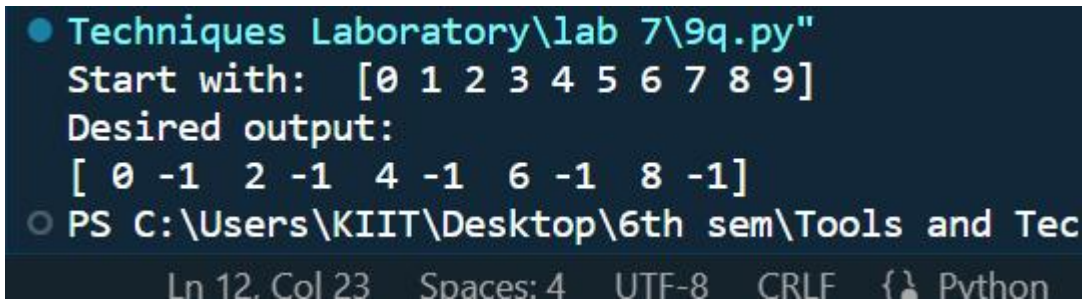
# Start with: exercise\_1 = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

# Desired output:

# [ 0, -1, 2, -1, 4, -1, 6, -1, 8, -1]

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)



```

● Techniques Laboratory\lab 7\9q.py"
Start with: [0 1 2 3 4 5 6 7 8 9]
Desired output:
[ 0 -1 2 -1 4 -1 6 -1 8 -1]
○ PS C:\Users\KIIT\Desktop\6th sem\Tools and Tec
Ln 12, Col 23 Spaces: 4 UTF-8 CRLF { } Python

```

**Source code**

(Paste here the source code)

```

import numpy as np
arr = np.array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])
print("Start with: ", arr)
print("Desired output: ")
arr[arr % 2 == 1] = -1
print(arr)

```

**Conclusion/Observation**

successfully compiled the code

**Program No: (10)****Program Title:**

# 10. WAP to create a 5X2 integer array from a range between 100 to 200 such that the difference between each element is 10

# Sample output:

# Creating 5X2 array using numpy.arange

# [[100 110]

# [120 130]

# [140 150]

# [160 170]

# [180 190]]

**Input/Output Screenshots:****RUN-1:**

(Paste here the screenshots of first run)



```
Techniques Laboratory (2007-2008)
Creating 5X2 array using numpy.arange
[[100 110]
 [120 130]
 [140 150]
 [160 170]
 [180 190]]
PS C:\Users\KIIT\Desktop\6th sem\Tools and Tech
```

**Source code**

(Paste here the source code)

```
import numpy as np
arr = np.arange(100, 200, 10).reshape(5, 2)
print("Creating 5X2 array using numpy.arange")
print(arr)
```

**Conclusion/Observation**

successfully compiled the code

