Q1

```
#!/usr/bin/env python3
import numpy as np


def find_factors(num):

    for i in range(1, num + 1):
        if num % i == 0:
            print(i, end=" , ")


def main():
    num = int(input("Enter the number to find the factors of: "))
    find_factors(num)


if __name__ == "__main__":
    main()
```

```
ugcse@prg28:~/Desktop/KaustavLABS4$ /usr/bin/env python3 "/home/ugcse/Desktop/KaustavLABS4/DS LAB/LAB 03/Practice_3_Programs/q1.py"
Enter the number to find the factors of: 1000
1 , 2 , 4 , 5 , 8 , 10 , 20 , 25 , 40 , 50 , 100 , 125 , 200 , 250 , 500 , 1000 , ugcse@prg28:~/Desktop/KaustavLABS4$
```

Q2

```
#!/usr/bin/env python3
import numpy as np


def main():
    elements = list([float(x) for x in input().split()])
    print(elements)

    nrows, ncols = [int(x) for x in input("Enter nrows and ncols: ").split()]

    elements_matrix = np.reshape(elements, [nrows, ncols])
    print(elements_matrix)
    print("column sum is: ", elements_matrix.sum(axis=0),
          "row sum is: ", elements_matrix.sum(axis=1))


if __name__ == "__main__":
    main()
```

```
ugcse@prg28:~/Desktop/KaustavLABS4$ /usr/bin/env python3 "/home/ugcse/Desktop/KaustavLABS4/DS LAB/LAB 03/Practice_3_Programs/q2.py"
1 2 3 4 5 6
[1.0, 2.0, 3.0, 4.0, 5.0, 6.0]
Enter nrows and ncols: 2 3
[[1. 2. 3.]
 [4. 5. 6.]]
column sum is:  [5. 7. 9.] row sum is:  [ 6. 15.]
```

Q3

```
#!/usr/bin/env python3
import numpy as np


class Q3():
    def a(self):
        my_list = [float(x) for x in input("Enter the list elements").split()]
        my_array = np.array(my_list)
        print(my_array)

    def b(self):
        my_tuple = tuple([float(x)
                          for x in input("Enter the list elements").split()])
```

```
        my_array2 = np.array(my_tuple)
        print(my_array2)

    def c(self):
        zero_matrix = np.zeros((3, 4))
        print(zero_matrix)

    def d(self):
        my_sequence = [int(x) for x in range(0, 21, 5)]
        print(my_sequence)

    def e(self):
        input_array = np.random.randn(3, 4)
        print("input_array", input_array)
        output_array = np.reshape(input_array, (2, 2, 3))
        print("output_array", output_array)

    def f(self):
        input_array = np.random.randn(3, 4)
        print("input_array", input_array)

        print("column sum is: ", input_array.sum(axis=0))
        print("row sum is: ", input_array.sum(axis=1))
        print("column max is: ", input_array.max(axis=0))
        print("row max is: ", input_array.max(axis=1))
        print("column min is: ", input_array.min(axis=0))
        print("row min is: ", input_array.min(axis=1))


if __name__ == "__main__":
    q3 = Q3()

    q3.a()
    q3.b()
    q3.c()
    q3.d()
    q3.e()
    q3.f()
```

```
ugcse@prg28:~/Desktop/KaustavLABS4$ /usr/bin/env python3 "/home/ugcse/Desktop/KaustavLABS4/DS LAB/LAB 03/Practice_3_Programs/q3.py"
Enter the list elements1 2 3 4 5 6
[1. 2. 3. 4. 5. 6.]
Enter the list elements1 2 3 4 5 6
[1. 2. 3. 4. 5. 6.]
[[0. 0. 0. 0.]
 [0. 0. 0. 0.]
 [0. 0. 0. 0.]]
[0, 5, 10, 15, 20]
input_array [[ 1.65032022  0.44434137  0.10672068  1.3301746 ]
 [-0.23928822 -0.42863671  0.13389096 -0.86761945]
 [ 1.09853226  1.0414242   0.48856971 -0.2866645 ]]
output_array [[[ 1.65032022  0.44434137  0.10672068]
  [ 1.3301746  -0.23928822 -0.42863671]]

 [[ 0.13389096 -0.86761945  1.09853226]
  [ 1.0414242   0.48856971 -0.2866645 ]]]
input_array [[-0.31357143 -0.91025545  0.41256339  1.48683461]
 [ 0.59112962  0.02216052  0.24842708  0.86350876]
 [ 1.69218499 -1.87841241 -0.58235347 -0.93386916]]
column sum is:  [ 1.96974318 -2.76650734  0.078637    1.41647421]
row sum is:  [ 0.67557112  1.72522598 -1.70245006]
column max is:  [1.69218499 0.02216052 0.41256339 1.48683461]
row max is:  [1.48683461 0.86350876 1.69218499]
column min is:  [-0.31357143 -1.87841241 -0.58235347 -0.93386916]
row min is:  [-0.91025545  0.02216052 -1.87841241]
```

Q4

```
#!/usr/bin/env python3
import numpy as np


def main():
    elements = list([float(x) for x in input().split()])
    print(elements)

    nrows, ncols = [int(x) for x in input("Enter nrows and ncols: ").split()]

    elements_matrix = np.reshape(elements, [nrows, ncols])
    print(elements_matrix, "Before transposing")

    elements_matrix = np.transpose(elements_matrix)
    print(elements_matrix, "After transposing")


if __name__ == "__main__":
    main()
```

```
ugcse@prg28:~/Desktop/KaustavLABS4$ /usr/bin/env python3 "/home/ugcse/Desktop/KaustavLABS4/DS LAB/LAB 03/Practice_3_Programs/q4.py"
1 2 3 4 5 6
[1.0, 2.0, 3.0, 4.0, 5.0, 6.0]
Enter nrows and ncols: 2 3
[[1. 2. 3.]
 [4. 5. 6.]] Before transposing
[[1. 4.]
 [2. 5.]
 [3. 6.]] After transposing
ugcse@prg28:~/Desktop/KaustavLABS4$
```

Q5

```
#!/usr/bin/env python3
import numpy as np


def main():
    input_array1 = np.random.randint(100, size=(3, 4))
    print("input_array1  \n", input_array1)

    input_array2 = np.random.randint(100, size=(3, 4))
    print("input_array2  \n", input_array2)

    print("sum \n", input_array1 + input_array2)


if __name__ == "__main__":
    main()
```

```
ugcse@prg28:~/Desktop/KaustavLABS4$ /usr/bin/env python3 "/home/ugcse/Desktop/KaustavLABS4/DS LAB/LAB 03/Practice_3_Programs/q5.py"
input_array1
 [[81 23 66 33]
 [81 91 98 83]
 [29 53 14 16]]
input_array2
 [[95 57 94 92]
 [42 55 15 30]
 [22 39 98 38]]
sum
 [[176  80 160 125]
 [123 146 113 113]
 [ 51  92 112  54]]
ugcse@prg28:~/Desktop/KaustavLABS4$
```

Q6

```
#!/usr/bin/env python3
import numpy as np


def main():
    input_array1 = np.random.randint(100, size=(3, 4))
    print("input_array1  \n", input_array1)

    input_array2 = np.random.randint(100, size=(3, 4))
    print("input_array2  \n", input_array2)

    print("sum \n", np.multiply(input_array1, input_array2))


if __name__ == "__main__":
    main()
```

```
ugcse@prg28:~/Desktop/KaustavLABS4$ /usr/bin/env python3 "/home/ugcse/Desktop/KaustavLABS4/DS LAB/LAB 03/Practice_3_Programs/q6.py"
input_array1
 [[22 23 29 45]
 [87 12 38 88]
 [63 74  4 33]]
input_array2
 [[84 64 80 50]
 [15 72 84 31]
 [83 30 48  6]]
sum
 [[1848 1472 2320 2250]
 [1305  864 3192 2728]
 [5229 2220  192  198]]
ugcse@prg28:~/Desktop/KaustavLABS4$
```