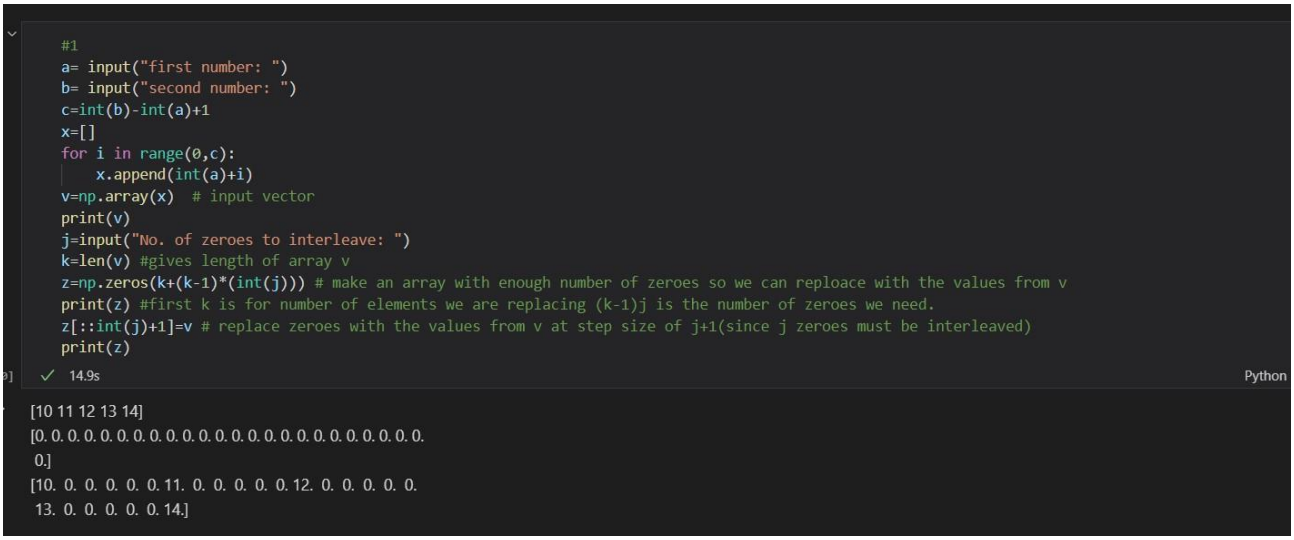
TASK –5

PYTHON (NUMPY AND PANDAS)

QUESTION 1: Consider the vector [10, 11, 12, 13, 14], how to build a new vector with 5 consecutive zeros interleaved between each value?

Solution:



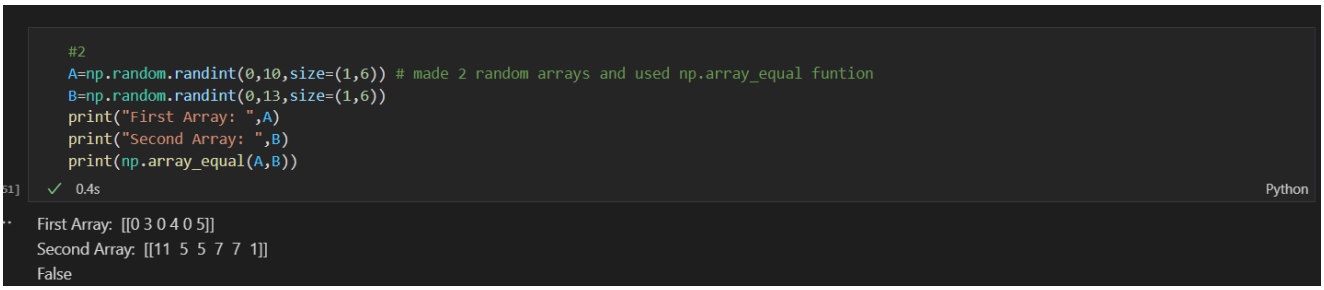
```
#1
a= input("first number: ")
b= input("second number: ")
c=int(b)-int(a)+1
x=[]
for i in range(0,c):
    x.append(int(a)+i)
v=np.array(x)  # input vector
print(v)
j=input("No. of zeroes to interleave: ")
k=len(v) #gives length of array v
z=np.zeros(k+(k-1)*(int(j))) # make an array with enough number of zeroes so we can reploace with the values from v
print(z) #first k is for number of elements we are replacing (k-1)j is the number of zeroes we need.
z[::int(j)+1]=v # replace zeroes with the values from v at step size of j+1(since j zeroes must be interleaved)
print(z)
```

✓ 14.9s Python

```
[10 11 12 13 14]
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.
 0.]
[10.  0.  0.  0.  0. 11.  0.  0.  0.  0. 12.  0.  0.  0.  0.  0.
 13.  0.  0.  0.  0. 14.]
```

QUESTION 2: Consider two random arrays A and B, check if they are equal

Solution:



```
#2
A=np.random.randint(0,10,size=(1,6)) # made 2 random arrays and used np.array_equal funtion
B=np.random.randint(0,13,size=(1,6))
print("First Array: ",A)
print("Second Array: ",B)
print(np.array_equal(A,B))
```

✓ 0.4s Python

```
First Array:  [[0 3 0 4 0 5]]
Second Array:  [[11  5  5  7  7  1]]
False
```

QUESTION 3: What is the result of the following expression?

print(0 * np.nan)

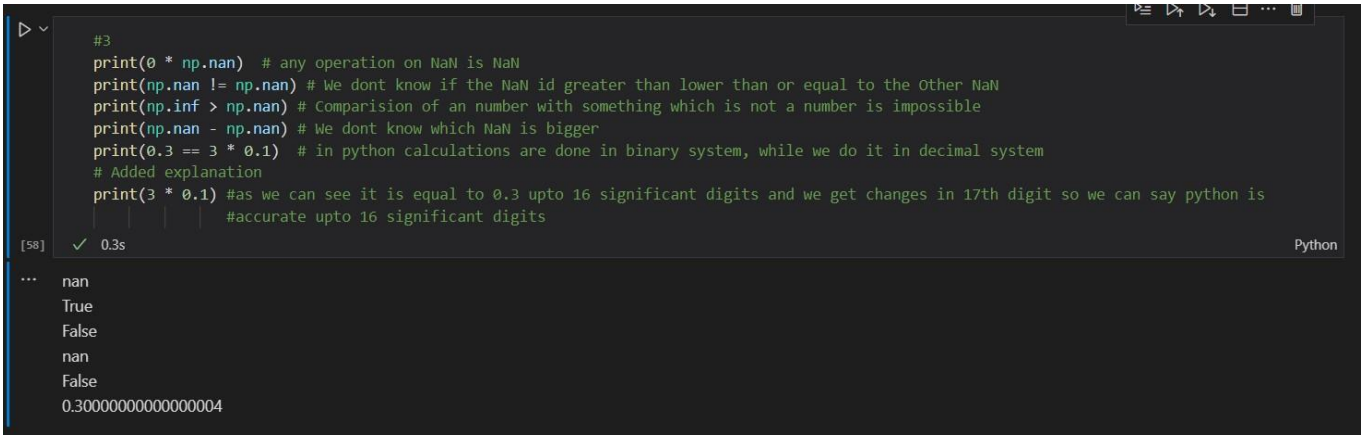
print(np.nan != np.nan)

print(np.inf > np.nan)

print(np.nan - np.nan)

print(0.3 == 3 * 0.1)

Solution:



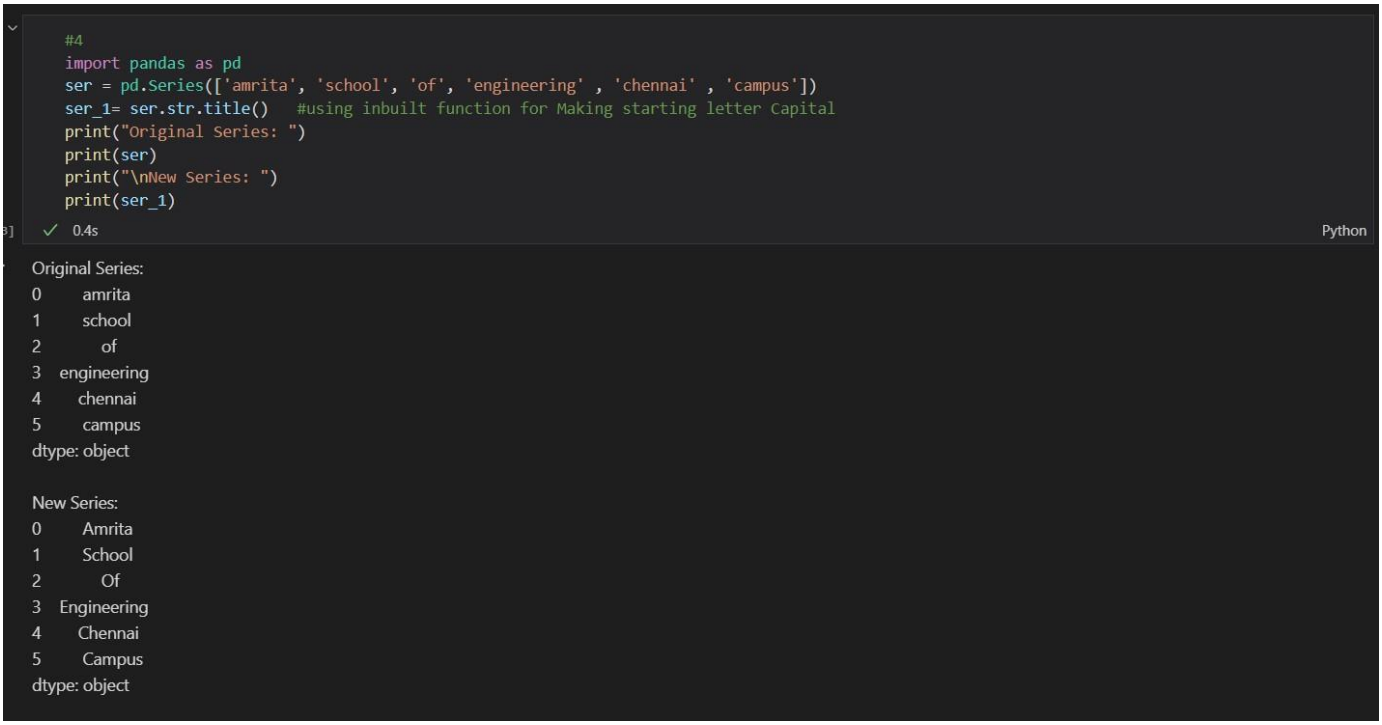
```
#3
print(0 * np.nan)  # any operation on NaN is NaN
print(np.nan != np.nan) # We dont know if the NaN id greater than lower than or equal to the Other NaN
print(np.inf > np.nan) # Comparision of an number with something which is not a number is impossible
print(np.nan - np.nan) # We dont know which NaN is bigger
print(0.3 == 3 * 0.1)  # in python calculations are done in binary system, while we do it in decimal system
# Added explanation
print(3 * 0.1) #as we can see it is equal to 0.3 upto 16 significant digits and we get changes in 17th digit so we can say python is
               #accurate upto 16 significant digits
```

[58] ✓ 0.3s Python

```
nan
True
False
nan
False
0.30000000000000004
```

QUESTION 4: Convert the first character of each element in a series to uppercase?

Solution:



```
#4
import pandas as pd
ser = pd.Series(['amrita', 'school', 'of', 'engineering' , 'chennai' , 'campus'])
ser_1= ser.str.title()   #using inbuilt function for Making starting letter Capital
print("Original Series: ")
print(ser)
print("\nNew Series: ")
print(ser_1)
```

✓ 0.4s Python

```
Original Series:
0       amrita
1       school
2           of
3   engineering
4      chennai
5       campus
dtype: object

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

QUESTION 5: Do any two Exercises using Numpy

1.addition of 2 numpy arrays

2.Multiplying a matrix

3.Identity Matrix

4.Array datatype conversion

5.Array re-dimensioning

6.Custom Sequence Generation

7.Getting the positions (indexes) where elements of 2 numpy arrays match

Solution:

```
#5
#1. Addition of 2 numpy arrays
print("1. Addition of 2 numpy arrays")
a=np.random.randint(0,9,size=(1,10))
b=np.random.randint(0,15,size=(1,10))
print("First array: ",a)
print("Second Array: ",b)
c=np.add(a,b)   #Using np.add funtion
print("Sum of Arrays: ",c)
#7 Getting matching array indices
print("7 Getting matching array indices")
g=np.array([1,2,3,6,9,8,0,6,1])
h=np.array([1,2,8,9,0,7,8,9,5])
print("First Array: ",g)
print("Second Array: ",h)
c_1=np.where(g==h)  #Using np.where to check for equal indicies
print("Equal indicies: ",c_1)
```

[57] ✓ 0.3s Python

```
1. Addition of 2 numpy arrays
First array:  [[7 0 8 4 0 2 6 2 5 4]]
Second Array:  [[ 9 12 14  8  8  1  0  3  7  9]]
Sum of Arrays:  [[16 12 22 12  8  3  6  5 12 13]]
7 Getting matching array indices
First Array:  [1 2 3 6 9 8 0 6 1]
Second Array:  [1 2 8 9 0 7 8 9 5]
Equal indicies:  (array([0, 1], dtype=int64),)
```