Write code to print characters from a string that are present at an even index number. For example, if your input string is "python", the output characters will be (p,t,o).

```
name=input("Enter your name \n")
modified_name= name[::2]
print(modified_name)
```

```
    Enter your name
    python
    pto
```

Write a function that calculates the average of the elements of the list:

elements = [5, 7, 4, 9, 2]

```
elements = [5, 7, 4, 9, 2]
su= sum(elements)
average=su/5
print(average)
```

```
    5.4
```

Write a code that takes a year as input and determines whether it is a leap year or not.

```
year = int(input('Enter a year: '))

if year % 400 == 0 and year % 100 == 0:
  print(year, 'is a leap year')
elif year % 4 == 0 and year % 100 != 0:
  print(year, 'is a leap year')
else:
  print(year, 'is not a leap year')
```

```
    Enter a year: 2004
    2004 is a leap year
```

Create a 1 dimensional numpy array of the first 50 even numbers. Convert this to a 10x5 numpy array and print its transpose.

```
import numpy as np

# Create 1D array of first 50 even numbers
arr_1d = np.arange(0, 100, 2)

# Reshape to 10x5 array
arr_2d = arr_1d.reshape(10, 5)

print(arr_2d)
print(arr_2d.T) #transpose of 10x5 array
```

```
    [[ 0  2  4  6  8]
     [10 12 14 16 18]
     [20 22 24 26 28]
     [30 32 34 36 38]
     [40 42 44 46 48]
```

```
 [50 52 54 56 58]
 [60 62 64 66 68]
 [70 72 74 76 78]
 [80 82 84 86 88]
 [90 92 94 96 98]]
[[ 0 10 20 30 40 50 60 70 80 90]
 [ 2 12 22 32 42 52 62 72 82 92]
 [ 4 14 24 34 44 54 64 74 84 94]
 [ 6 16 26 36 46 56 66 76 86 96]
 [ 8 18 28 38 48 58 68 78 88 98]]
```

Consider the 10x5 array created in the previous question. Find the mean of each row and column. You may use in built functions in order to have a more optimal solution

```python
#Mean of column
mean_of_column = arr_2d.mean(axis=0)
print ("Mean of column",mean_of_column)

#Mean of row
mean_of_row = arr_2d.mean(axis=1)
print ("Mean of row",mean_of_row)
```

```
Mean of column [45. 47. 49. 51. 53.]
Mean of row [ 4. 14. 24. 34. 44. 54. 64. 74. 84. 94.]
```

Colab paid products - Cancel contracts here