Q1. Can you create a programme or function that employs both positive and negative indexing? Isthere any repercussion if you do so?

```
Yes we can create a programme or function that employs both positive and negative indexing
there is no repercussion if you do so
```

Q2. What is the most effective way of starting with 1,000 elements in a Python list? Assume that allelements should be set to the same value.

```
value=1
l=[value]*1000

```

Q3. How do you slice a list to get any other part while missing the rest? (For example, suppose youwant to make a new list with the elements first, third, fifth, seventh, and so on.)

In [4]:

```
l=[1,2,3,3,4,5,6,6,7,8,0]
new=l[1::2]
print(new)
```

```
[1, 3, 4, 6, 7, 0]
```

Q4. Explain the distinctions between indexing and slicing.

```
indexing is used to acces single elemtnt or characters
Slicing is used to access elements or character of certain range
```

Q5. What happens if one of the slicing expression's indexes is out of range?

```
It will give the string from guven strting index to end of the string
```

Q6. If you pass a list to a function, and if you want the function to be able to change the values of thelist—so that the list is different after the function returns—what action should you avoid? ¶

```
We should avoid reassigning the entire list to a new object within the function
```

Q7. What is the concept of an unbalanced matrix?

```
The matrix not having same number of rows and columns
```

Q8. Why is it necessary to use either list comprehension or a loop to create arbitrarily large matrices?

```
With list comprehension or loops, you can iterate over rows and columns of the matrix and generate the elements
on the fly, without having to predefine the entire matrix structure.

```