

Experiment 2 e)

Aim: Implementing Repetition operation on Strings, List, Tuple, Set, Dictionary.

Theory:

Repetition operator (*):

- The repetition operator * will make multiple copies of a particular object and combine them together.
- When * is used with an integer it performs multiplication but with [list](#), [tuple](#) or [strings](#) it performs a repetition.
- All [mutable and immutable data types](#) support repetition operators.
- **if we use a repetition operator with a set or dictionary it will raise TypeError.**
- **if we give a repetition value less than zero or equal to zero then an empty sequence will be generated.**

Repetition operator on String, List and Tuple:

#Repetition operator on String

```
Result='Python For DS '
```

```
a = Result*3
```

```
print(a)
```

Output:

```
Python For DS Python For DS Python For DS
```

#Repetition operator on list

```
Result=[1,2,3,4]*3
```

```
print(Result)
```

Output:

```
[1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4]
```

#Repetition operator on tuple

```
Result=(1,2,3,4)*3
```

```
print(Result)
```

Output:

```
[1, 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4]
```

#Repetition operator on list with 0

```
Result=[1,2,3,4]*0
```

```
print(Result)
```

Output:

```
[ ]
```

Repetition operator on list with less than zero

```
Result=[1,2,3,4]*-5  
print(Result)
```

Output:

```
[ ]
```

Repetition operator on Set dictionary):

➤ To repeat the set or dictionary **repeat() method** is used.

repeat() method

- **itertools** module is imported to use the repeat method.
- **repeat()** method returns object over again and again until times limit is reached.
- if we do not specify time argument then it will repeat endlessly.

#Set repetition

```
import itertools  
MySet={1,2,3,4}  
result=itertools.repeat(MySet,times=3)  
print(list(result))
```

Output:

```
{1, 2, 3, 4}, {1, 2, 3, 4}, {1, 2, 3, 4}
```

#Dictionary repetition

```
import itertools  
MyDict={"Name":"John","age":27}  
result=itertools.repeat(MyDict,times=3)  
print(list(result))
```

Output:

```
{'Name': 'John', 'age': 27}, {'Name': 'John', 'age': 27}, {'Name': 'John', 'age': 27}
```

Conclusion:

Hence implemented Repetition operation for String, List, Tuple, Set and Dictionary.