

Compute and store the squares of each index of array in new array

In [3]:

```
def simple_approch(fields):
    sqr = []

    for i in fields:
        sqr.append((i*i))

    print(sqr)

def comprehension_approch(fields):
    sqr = [i*i for i in fields] #Comprehensive way to perform above task

    print(sqr)

fields = [1,2,3,4,5]

simple_approch(fields)
comprehension_approch(fields)
```

```
[1, 4, 9, 16, 25]
```

```
[1, 4, 9, 16, 25]
```

Create a Dictionary against field values using comprehensive approach

In [16]:

```
def comprehension_approch(fields, val):
    dicc = {i:val for i in fields} # Each field value will act as key and against it val will be set as value

    print(dicc)

fields = [1,2,3,4,5]
comprehension_approch(fields, 5)
```

```
{1: 5, 2: 5, 3: 5, 4: 5, 5: 5}
```

Conditional Comprehensive approach

In [17]:

```
#Build Dictionary where keys are field values and values are key + 10, This must be done if key > 2

fields = [1,2,3,4,5]
dicc = {i:i+10 for i in fields if i > 2}
print(dicc)
```

```
{3: 13, 4: 14, 5: 15}
```