(3.8) bytes & bytearray

bytes data type represens a group of byte numbers just like an array

In [2]:
```
x = [10,20,30,40]

b = bytes(x)

print(b)

print(type(b))

print(b[0])

print(b[-1])
```

```
b'\n\x14\x1e('
<class 'bytes'>
10
40
```

In [3]:
```
for i in b : print(i)
```

```
10
20
30
40
```

Conclusion 1:

The only allowed values for byte data type are 0 to 255. By mistake if we are trying to
provide any other values then we will get value error.

Conclusion 2:

Once we creates bytes data type value, we cannot change its values,otherwise we will get
TypeError.

In [4]:
```
x = [10,20,30,40]
b = bytes(x)
b[0] = 100
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
C:\Users\PANKAJ~1\AppData\Local\Temp/ipykernel_6384/1789588826.py in <module>
      1 x = [10,20,30,40]
      2 b = bytes(x)
----> 3 b[0] = 100

TypeError: 'bytes' object does not support item assignment
```

In [5]:
```
x = [10,20,300,40]
b = bytes(x)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
```

```
C:\Users\PANKAJ~1\AppData\Local\Temp/ipykernel_6384/3810128340.py in <module>
      1 x = [10,20,300,40]
----> 2 b = bytes(x)

ValueError: bytes must be in range(0, 256)
```

(3.8.1) bytearray

bytearray is exactly same as bytes data type except that its elements can be modified

In [4]:

```
x = [10,20,30,40]
b = bytearray(x)
print(b)
b[0] = 100

for i in b : print(i)
```

```
bytearray(b'\n\x14\x1e(')
100
20
30
40
```

In [ ]: