In [2]:
```
import torch
```

In [4]:
```
torch.__version__
```

Out[4]:
```
'2.4.1'
```

In [4]:
```
torch.__version__
```

Out[4]:
```
'2.4.1'
```

In [6]:
```
import numpy as np
```

In [9]:
```
lst = [3,4,5,6]
arr = np.array(lst)
```

In [10]:
```
arr
```

Out[10]:
```
array([3, 4, 5, 6])
```

In [12]:
```
#convert numpy to pytorch tensors (Tensors are nothing but replacement of numpy to use the power of GPU and it is a generalized form of vector and matrices which we can easily understand as multidimensional array)

tensors = torch.from_numpy(arr)
tensors
```

Out[12]:
```
tensor([3, 4, 5, 6], dtype=torch.int32)
```

In [13]:
```
#indexing in tensors its similar to numpy,in array it starts from 0,1,2,....
tensors[ :2]
```

Out[13]:
```
tensor([3, 4], dtype=torch.int32)
```

In [15]:
```
tensors[0:4]
```

Out[15]:
```
tensor([3, 4, 5, 6], dtype=torch.int32)
```

In [17]:
```
#disadvantages of from_numpy it uses same location as the array so any changeswe make in tensors also reflected in original arrar
tensors[3]=100
tensors
```

Out[17]:
```
tensor([  3,   4,   5, 100], dtype=torch.int32)
```

In [18]:
```
arr
```

Out[18]:
```
array([  3,   4,   5, 100])
```

In [19]:
```
#To prevent this error we use torch.tensor instead of from_numpy
tensors_arr = torch.tensor(arr)
tensors_arr
```

Out[19]:
```
tensor([  3,   4,   5, 100], dtype=torch.int32)
```

In [26]:
```
tensors_arr[3]=120
print(tensors_arr)
print(arr)
```

```
tensor([  3,   4,   5, 120], dtype=torch.int32)
[  3   4   5 100]
```

In [27]:
```
#Arithmetic Operations
a = torch.tensor([3,4,5], dtype =torch.float)
b = torch.tensor([6,7,8], dtype = torch.float)
a+b
```

Out[27]:
```
tensor([ 9., 11., 13.])
```

In [28]:
```
torch.add(a,b)
```

Out[28]:
```
tensor([ 9., 11., 13.])
```

In [33]:
```
c = torch.zeros(3)
torch.add(a,b,out=c)
```

Out[33]:
```
tensor([ 9., 11., 13.])
```

In [36]:
```
a = torch.tensor([7,8,9],dtype=torch.float)
b=torch.tensor([10,12,14],dtype=torch.float)
```

In [37]:
```
torch.add(a,b).sum()
```

Out[37]:
```
tensor(60.)
```

In [38]:
```
#Dot product and Mult Operations
x = torch.tensor([8,12,13],dtype=torch.float)
y = torch.tensor([13,19,21],dtype=torch.float)
```

In [39]:
```
x.mul(y)
```

Out[39]:
```
tensor([104., 228., 273.])
```

In [40]:
```
x.dot(y)
```

Out[40]:
```
tensor(605.)
```

In [41]:
```
#Matrix Multiplication
x = torch.tensor([[1,4,2],[1,5,5]],dtype=torch.float)
y= torch.tensor([[5,11],[8,6],[9,11]],dtype=torch.float)
```

In [45]:
```
torch.matmul(x,y)
```

Out[45]:
```
tensor([[55., 57.],
        [90., 96.]])
```

In [46]:
```
torch.mm(x,y)
```

Out[46]:
```
tensor([[55., 57.],
        [90., 96.]])
```

In [47]:
```
x@y
```

Out[47]:
```
tensor([[55., 57.],
        [90., 96.]])
```