```
#%% md
# PyTorch Tensor
#%%
import torch
#%% md
### Tensor Allocation
#%%
ft = torch.FloatTensor([[1, 2],
                        [3, 4]])
ft
#%%
lt = torch.LongTensor([[1, 2],
                       [3, 4]])
lt
#%%
bt = torch.ByteTensor([[1, 0],
                       [0, 1]])
bt
#%%
x = torch.FloatTensor(3, 2)
x
#%% md
### NumPy Compatibility
#%%
import numpy as np

# Define numpy array.
x = np.array([[1, 2],
              [3, 4]])
print(x, type(x))
#%%
x = torch.from_numpy(x)
print(x, type(x))
#%%
x = x.numpy()
print(x, type(x))
#%% md
### Tensor Type-casting
#%%
ft.long()
#%%
```

```
lt.float()
#%%
torch.FloatTensor([1, 0]).byte()
#%% md
### Get Shape
#%%
x = torch.FloatTensor([[[1, 2],
                        [3, 4]],
                       [[5, 6],
                        [7, 8]],
                       [[9, 10],
                        [11, 12]]])
#%% md
##### - Get tensor shape.
#%%
print(x.size())
print(x.shape)
#%% md
##### - Get number of dimensions in the tensor.
#%%
print(x.dim())
print(len(x.size()))
#%% md
##### - Get number of elements in certain dimension
    of the tensor.
#%%
print(x.size(1))
print(x.shape[1])
#%% md
##### - Get number of elements in the last
   dimension.
#%%
print(x.size(-1))
print(x.shape[-1])
```