

Implement a program for matrix multiplication for the given matrix

Steps:

1. Check the device whether GPU is available or not in the machine and set the GPU if it is available

2. Create tensor x_tensor with the values

1	2	3
4	5	6
7	8	9

3. Create tensor y_tensor with the values:

9	8	7
6	5	4
3	2	1

4. Multiply the created matrices x_tensor and y_tensor and store in z_tensor.

Hint: use '@' operator of pytorch for dot product

```
In [2]: import torch

# Set device based on availability of CUDA
device = 'cuda' if torch.cuda.is_available() else 'cpu'
print(f"Using device: {device}")

#Create x_tensor and y_tensor and assign to the selected device.
### BEGIN SOLUTION
x = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]
y = [[9,8,7], [6,5,4], [3,2,1]]

### END SOLUTION
# Convert to PyTorch tensors and move to the chosen device
x_tensor = torch.tensor(x, dtype=torch.float).to(device)
y_tensor = torch.tensor(y, dtype=torch.float).to(device)

# Apply @ operator for matrix multiplication on x_tensor and y_tensor and store
### BEGIN SOLUTION
z_tensor=x_tensor@y_tensor
z_tensor
### END SOLUTION
```

Using device: cuda

```
[NVSHARE][WARN]: Couldn't open file /var/run/secrets/kubernetes.io/serviceaccount/namespace to read Pod namespace  
[NVSHARE][INFO]: Successfully initialized nvshare GPU  
[NVSHARE][INFO]: Client ID = ae4f58de5ee0db65
```

```
Out[2]: tensor([[ 30.,  24.,  18.],  
               [ 84.,  69.,  54.],  
               [138., 114.,  90.]], device='cuda:0')
```