Core components of pytorch

1. Tensor library for efficient computing
2. Automatic differentiation engine – utilities to differentiate computations automatically
3. Deep learning library

Defining Deep learning:

1. If cuda compatible gpu is found , torch for gpu will be automatically installed.
2. pip install torch torchvision torchaudio --index-url <https://download.pytorch.org/whl/cu126> [126 is cuda 12.6]
3. nvidia-smi =>The **nvidia-smi** (NVIDIA System Management Interface) command displays information about your **GPU**, including driver version, CUDA version, memory usage, and running processes.
4. Float 32 is generated by default and its optimized for GPU
5. Most operations are similar to GPU
6. If we carry out the computations in Pytorch , it will build a computational graph internally by default if one of its terminal nodes has requires\_grad =True
7. By default ..after calculating the gradient , the computational graph is destroyed but here using retain\_graph=True, we retain the graph as we do multiple grad calculations