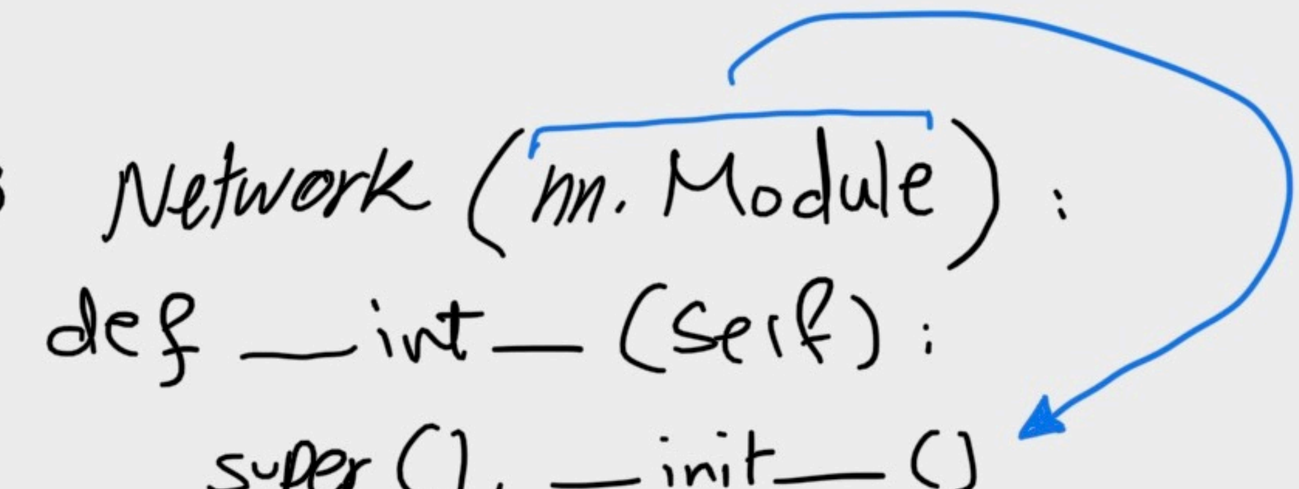
Using nn module to build neural nets:

```
from torch import nn

class Network(nn.Module):
    def __int__(self):
        super().__init__()
        self.hidden = nn.Linear(784, 256)
        self.output = nn.Linear(256, 10)
```



- ★ self.sigmoid = nn.Sigmoid
- ★ self.softmax = nn.Softmax(dim=1)

```
    def forward(self, x):
        x = self.hidden(x)
        x = self.sigmoid(x)
        x = self.output(x)
        x = self.softmax(x)
        return x
```

For making things easier: import torch.nn.functional as F

then we can remove self.sigmoid and self.softmax from the class above.

```
def forward(self, x):
    x = F.sigmoid(self.hidden(x))
    x = F.softmax(self.output(x), dim=1)
    return x
```