The Code Thus Far..

In this lesson, we will take a look at the code we have created so far.

Let's take a breath and pause to check what the code for the neural network class we're building up looks like. It should look something like the following.

```
# neural network class definiti
class neuralNetwork:


    # initialise the neural net
    def __init__(self, inputnod
        # set number of nodes i
        self.inodes = inputnode
        self.hnodes = hiddennod
        self.onodes = outputnod

        # link weight matrices,
        # weights inside the ar
        # w11 w21
        # w12 w22 etc
        self.wih = numpy.rando
        self.who = numpy.rando

        # learning rate
        self.lr = learningrate

        # activation function i
        self.activation_functi


        pass


    # train the neural network
    def train():
        pass

```

That is just the class, aside from that we should be importing the `numpy` and `scipy.special` modules right at the top of the code in the cell below:

```
import numpy
# scipy.special for the sigmoid function expit()
import scipy.special

```

It is worth briefly noting the `query()` function only needs the `input_list`. It doesn't need any other input.

That's good progress, and now we look at the missing piece, the `train()` function. Remember there are two phases to training, the first is calculating the output just as `query()` does it, and the second part is backpropagating the errors to inform how the link weights are refined.