* First phase : creating the model
  + Neural networks is defined by a set of layers inside Sequentiel :  
    model= keras.Sequential([keras.layers.Dense(units=1 , input\_shape=[1]) , Dense() , ….. ])
  + Dense defines a defined kind of layer with connected neurons (so neural networks is a set of layers)
  + Dense has two main args :
    - units : the number of neurons inside the layer
    - Input\_shape=[1] : it means that the expected input for this layer is just an array with a single element
* Second phase is comopiling the model
  + Model.compile() needs two arguments
    - Loss ( ‘mean\_squared\_error’ ) : it defines the function that the model gonna use to calculate the loss ( eroor) between the expected label and the predicted one ( y and y^)
    - Optimizer ( ‘sgd’ ) : The role of the optimizer is to take the calculated loss and measure and Generates a new and improved guess
* Third phase : feeding the model with the data and doing the fitting ( X and Y )
  + Model.fit() needs two arguments
    - X : is the inputs
    - Y is the output ( labels )
    - Epochs : the number of the predictions attempts by our neural network to do the predictions
* Fourth Phase : Predicting a Unknown Y for a defined x
  + Model.predict() needs one argument
    - The input\_shape form argument : it’s our given input to predict it’s output through our Neural network