**Assignment No: 1**

**Problem Statement:-**

Implement Feedforward Neural Networks (FNN) in Python using **Keras** and **TensorFlow**.

**Theory:-**

A **Feedforward Neural Network (FNN)** is one of the simplest forms of artificial neural networks, where the data flows in one direction—from input nodes, through hidden layers, to output nodes. It does not form cycles or loops (unlike Recurrent Neural Networks).

* **Layers**: FNNs consist of multiple layers: input, hidden, and output layers.
* **Activation Functions**: Non-linear functions like **ReLU**, **Sigmoid**, and **Softmax** are used to introduce non-linearity into the network, enabling it to learn complex patterns.

**Methodology:-**

1. **Data Preparation**:
   * Use a dataset such as **MNIST** for handwritten digit recognition, which consists of 28x28 pixel images.
   * Preprocess the data by normalizing pixel values to the range [0, 1].
2. **Model Architecture**:
   * Define an input layer corresponding to the number of features (e.g., 784 for 28x28 images).
   * Add one or more hidden layers with a set number of neurons, using the **ReLU** activation function.
   * Define an output layer with the number of classes (for classification tasks, use **Softmax** activation).
3. **Implementation in Keras**:
   * Use **Sequential** API to build the feedforward network.
   * Compile the model using an optimizer (like **Adam**) and a loss function (e.g., **categorical\_crossentropy** for classification).
4. **Training and Evaluation**:
   * Train the model on training data using the fit() function and validate on a test set.
   * Evaluate the model's accuracy using the evaluate() function.

**Conclusion:-**

We successfully implemented a feedforward neural network using Keras and TensorFlow, achieving good accuracy on a basic classification task like MNIST digit classification.