

딥러닝 코딩하기

```
<!DOCTYPE html>
<html>

<head>
  <title>TensorFlow.js Tutorial - boston housing </title>

  <!-- Import TensorFlow.js -->
  <script
src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs@1.0.0/dist/tf.min.js"
></script>
    <script src="https://cdn.jsdelivr.net/npm/@tensorflow/tfjs-vis">
</script>
    <script src="10.3.js"></script>
</head>

<body>
  <script>
    /*
      var 보스톤_원인 = [

[0.00632,18,2.31,0,0.538,6.575,65.2,4.09,1,296,15.3,396.9,4.98],

[0.02731,0,7.07,0,0.469,6.421,78.9,4.9671,2,242,17.8,396.9,9.14]
    ];
    var 보스톤_결과 = [
      [24],
      [21.6]
    ];
    */

    // 1. 과거의 데이터를 준비합니다.
    var 원인 = tf.tensor(보스톤_원인);
    var 결과 = tf.tensor(보스톤_결과);

    // 2. 모델의 모양을 만듭니다.
    var X = tf.input({ shape: [13] }); // 입력 층
```

```

        var H1 = tf.layers.dense({ units: 13, activation: 'relu'
    }).apply(X); // 은닉층 추가 relu : activation function
        var H2 = tf.layers.dense({ units: 13, activation: 'relu'
    }).apply(H1); // 은닉층 추가
        var Y = tf.layers.dense({ units: 1 }).apply(H2); // 출력 층
        var model = tf.model({ inputs: X, outputs: Y });
        var compileParam = { optimizer: tf.train.adam(), loss:
tf.losses.meanSquaredError }
        model.compile(compileParam);
        tfvis.show.modelSummary({ name: '요약', tab: '모델' }, model);

        // 3. 데이터로 모델을 학습시킵니다.
        //         var fitParam = {epochs: 100}
        var _history = [];
        var fitParam = {
            epochs: 100,
            callbacks: {
                onEpochEnd:
                    function (epoch, logs) {
                        console.log('epoch', epoch, logs, 'RMSE=>',
Math.sqrt(logs.loss));
                        _history.push(logs);
                        tfvis.show.history({ name: 'loss', tab: '역사' },
_history, ['loss']);
                    }
            }
        } // loss 추가 예제
        model.fit(원인, 결과, fitParam).then(function (result) {

            // 4. 모델을 이용합니다.
            // 4.1 기존의 데이터를 이용
            var 예측한결과 = model.predict(원인);
            예측한결과.print();

        });

        // 4.2 새로운 데이터를 이용
        // var 다음주온도 = [15,16,17,18,19]
        // var 다음주원인 = tf.tensor(다음주온도);
        // var 다음주결과 = model.predict(다음주원인);
        // 다음주결과.print();
    </script>
</body>

</html>

```

-
- 은닉층은 여러개가 올 수 있다.