```
import tensorflow as tf
import numpy as np


np.random.seed(0)
x_train = np.random.rand(100, 8)
y_train = np.random.randint(low = 0, high =4, size = 100)
print(x_train.shape)
print(x_train[0])
print(y_train)
```

```
(100, 8)
[0.5488135  0.71518937 0.60276338 0.54488318 0.4236548  0.64589411
 0.43758721 0.891773  ]
[0 1 0 3 1 2 2 3 0 1 2 1 3 3 1 1 3 3 0 2 0 1 2 0 2 0 1 3 2 1 1 3 3 3 2 3 2
 3 0 0 1 0 3 2 2 3 2 1 0 3 0 0 3 3 0 3 0 0 0 3 3 1 0 0 0 1 0 1 1 3 1 3 2 0
 2 0 1 3 1 0 0 1 0 0 2 1 2 1 2 3 0 0 1 1 1 2 1 0 1 0]
```

```
def create_model():
  model = tf.keras.models.Sequential([
    tf.keras.layers.Dense(8, input_shape = (8,), activation='relu', use_bias= True ),#filter size 8
    tf.keras.layers.Dense(4, use_bias=True)         # 위에 0123 가지수가 4개이므로 4
  ])
  model.compile(
      #optimization = 'SGD',
      optimizer = tf.keras.optimizers.SGD(learning_rate= 0.01),
      loss = tf.keras.losses.SparseCategoricalCrossentropy(from_logits= True),
      metrics = ['accuracy']
  )
  return model



model1 = create_model()
model1.summary()
model1.fit(x_train, y_train, batch_size = 1, epochs = 10)
```

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
dense_4 (Dense)              (None, 8)                 72
_________________________________________________________________
dense_5 (Dense)              (None, 4)                 36
=================================================================
Total params: 108
Trainable params: 108
Non-trainable params: 0
_________________________________________________________________
Epoch 1/10
100/100 [==============================] - 0s 793us/step - loss: 1.4522 - accuracy: 0.2300
Epoch 2/10
100/100 [==============================] - 0s 771us/step - loss: 1.3928 - accuracy: 0.2700
Epoch 3/10
100/100 [==============================] - 0s 783us/step - loss: 1.3700 - accuracy: 0.3300
Epoch 4/10
```

```
100/100 [==============================] - 0s 870us/step - loss: 1.3568 - accuracy: 0.3300
Epoch 5/10
100/100 [==============================] - 0s 776us/step - loss: 1.3453 - accuracy: 0.3500
Epoch 6/10
100/100 [==============================] - 0s 762us/step - loss: 1.3400 - accuracy: 0.3700
Epoch 7/10
100/100 [==============================] - 0s 882us/step - loss: 1.3323 - accuracy: 0.3700
Epoch 8/10
100/100 [==============================] - 0s 797us/step - loss: 1.3285 - accuracy: 0.3500
Epoch 9/10
100/100 [==============================] - 0s 732us/step - loss: 1.3247 - accuracy: 0.3600
Epoch 10/10
100/100 [==============================] - 0s 850us/step - loss: 1.3186 - accuracy: 0.3800
<tensorflow.python.keras.callbacks.History at 0x7fd0c65aefd0>
```