

```
import tensorflow as tf
import numpy as np
```

```
celsius = np.array([-40, -10, 0, 8, 15, 22, 38], dtype=float)
fahrenheit = np.array([-40, 14, 32, 46, 59, 72, 100], dtype=float)
```

```
#capa = tf.keras.layers.Dense(units=1, input_shape=[1])
#modelo = tf.keras.Sequential([capa])

oculta1 = tf.keras.layers.Dense(units=3, input_shape=[1])
oculta2 = tf.keras.layers.Dense(units=3)
salida = tf.keras.layers.Dense(units=1)
modelo = tf.keras.Sequential([oculta1, oculta2, salida])
```

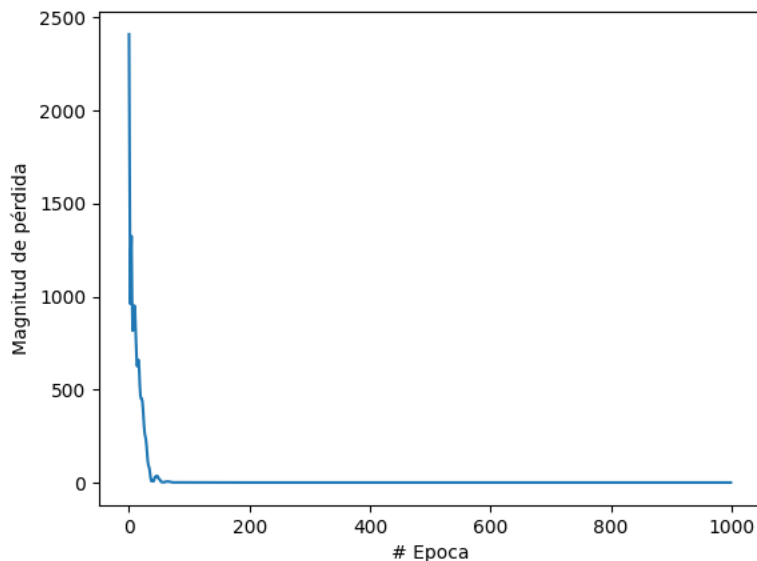
```
modelo.compile(
    optimizer=tf.keras.optimizers.Adam(0.1),
    loss='mean_squared_error'
)
```

```
print("Comenzando entrenamiento...")
historial = modelo.fit(celsius, fahrenheit, epochs=1000, verbose=False)
print("Modelo entrenado!")
```

```
Comenzando entrenamiento...
Modelo entrenado!
```

```
import matplotlib.pyplot as plt
plt.xlabel("# Epoca")
plt.ylabel("Magnitud de pérdida")
plt.plot(historial.history["loss"])
```

```
[<matplotlib.lines.Line2D at 0x7f539bed6cb0>]
```



```
print("Hagamos una predicción!")
resultado = modelo.predict([100.0])
print("El resultado es " + str(resultado) + " fahrenheit!")
```

```
Hagamos una predicción!
1/1 [=====] - 0s 190ms/step
El resultado es [[211.74742]] fahrenheit!
```

```
print("Variables internas del modelo")
#print(capa.get_weights())
print(oculta1.get_weights())
print(oculta2.get_weights())
print(salida.get_weights())
```

```
Variables internas del modelo
[array([[-0.21098563, 0.27083352, 0.8479363 ]], dtype=float32), array([[-3.801898, 3.762913, 3.9242435], dtype=float32)]
```

```
[array([[ 0.34652913,  0.75701517, -0.8932532 ],
        [-1.3232809 , -0.9518553 , -0.0162547 ],
        [-0.16120557,  0.16377413,  1.2334045 ]], dtype=float32), array([-3.828922 , -3.818286 ,  3.9366963], dtype=float32)]
[array([-0.9673054],
        [-0.8138459],
        [ 0.8305971]], dtype=float32), array([3.6441126], dtype=float32)]
```

[Productos pagados de Colab](#) - [Cancela los contratos aquí](#)

✓ 0 s se ejecutó 11:14

