

```

from google.colab import auth
import gspread
from google.auth import default
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import MinMaxScaler
from keras.models import Sequential
from keras.layers import Dense
from sklearn.preprocessing import StandardScaler

```

```

auth.authenticate_user()
creds, _ = default()
gc = gspread.authorize(creds)

```

```

worksheet = gc.open('student_data').sheet1
rows = worksheet.get_all_values()
df = pd.DataFrame(rows[1:], columns=rows[0])
print(df.shape)

```

(20, 2)

```

worksheet=gc.open('student_data').sheet1
rows = worksheet.get_all_values()

```

```

dataset1 = pd.DataFrame(rows[1:], columns=rows[0])
dataset1 = dataset1.astype({'Input': 'int'})
dataset1 = dataset1.astype({'output': 'int'})

```

```
dataset1.head()
```

	Input	output
0	1	11
1	2	21
2	3	31
3	4	41
4	5	51

```

X = dataset1[['Input']].values
y_train = dataset1[['output']].values

```

```
#x
```

```
from sklearn.model_selection import train_test_split
```

```
X_train, X_test, y_train, y_test = train_test_split(X, y_train, test_size=0.33, random_state=33)
```

```
scaler = MinMaxScaler()
```

```
scaler.fit(X_train)
```

```

MinMaxScaler
MinMaxScaler()

```

```
X_train1= scaler.transform(X_train)
```

```

ai_brain = Sequential([
Dense(8,activation='relu'),
Dense(4,activation='relu'),
Dense(1)
])

```

```
ai_brain.compile(optimizer= 'rmsprop', loss="mse")
```

```
ai_brain.fit(X_train1,y_train,epochs=2000)
```

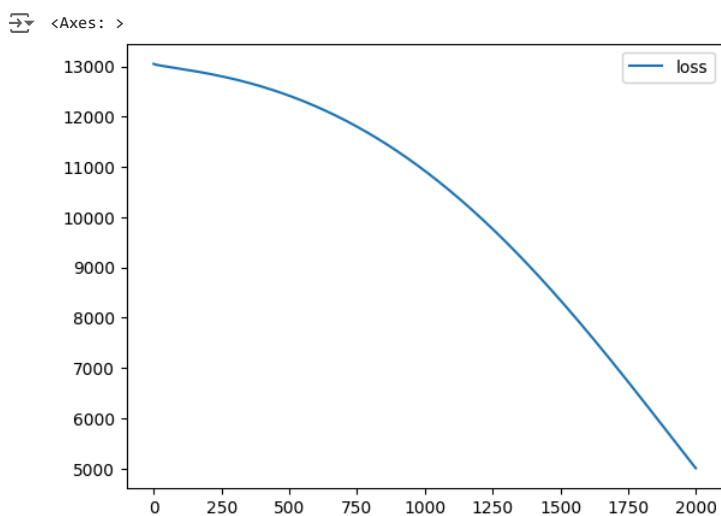
```

1/1 ----- 0s 149ms/step - loss: 5191.0107
Epoch 1974/2000
1/1 ----- 0s 151ms/step - loss: 5184.6997
Epoch 1975/2000
1/1 ----- 0s 158ms/step - loss: 5177.7837
Epoch 1976/2000
1/1 ----- 0s 125ms/step - loss: 5170.8682
Epoch 1977/2000
1/1 ----- 0s 147ms/step - loss: 5163.9536
Epoch 1978/2000
1/1 ----- 0s 87ms/step - loss: 5157.0400
Epoch 1979/2000
1/1 ----- 0s 147ms/step - loss: 5150.1274
Epoch 1980/2000
1/1 ----- 0s 55ms/step - loss: 5143.2144
Epoch 1981/2000
1/1 ----- 0s 78ms/step - loss: 5136.3027
Epoch 1982/2000
1/1 ----- 0s 128ms/step - loss: 5129.3926
Epoch 1983/2000
1/1 ----- 0s 60ms/step - loss: 5122.4824
Epoch 1984/2000
1/1 ----- 0s 34ms/step - loss: 5115.5732
Epoch 1985/2000
1/1 ----- 0s 59ms/step - loss: 5108.6641
Epoch 1986/2000
1/1 ----- 0s 64ms/step - loss: 5101.7559
Epoch 1987/2000
1/1 ----- 0s 33ms/step - loss: 5094.8496
Epoch 1988/2000
1/1 ----- 0s 34ms/step - loss: 5087.9434
Epoch 1989/2000
1/1 ----- 0s 33ms/step - loss: 5081.0386
Epoch 1990/2000
1/1 ----- 0s 43ms/step - loss: 5074.1333
Epoch 1991/2000
1/1 ----- 0s 55ms/step - loss: 5067.2300
Epoch 1992/2000
1/1 ----- 0s 58ms/step - loss: 5060.3276
Epoch 1993/2000
1/1 ----- 0s 56ms/step - loss: 5053.4253
Epoch 1994/2000
1/1 ----- 0s 60ms/step - loss: 5046.5239
Epoch 1995/2000
1/1 ----- 0s 59ms/step - loss: 5039.6245
Epoch 1996/2000
1/1 ----- 0s 63ms/step - loss: 5032.7261
Epoch 1997/2000
1/1 ----- 0s 61ms/step - loss: 5025.8276
Epoch 1998/2000
1/1 ----- 0s 35ms/step - loss: 5018.9302
Epoch 1999/2000
1/1 ----- 0s 54ms/step - loss: 5012.0347
Epoch 2000/2000
1/1 ----- 0s 38ms/step - loss: 5005.1396
<keras.src.callbacks.history.History at 0x7a94437401f0>

```

```
loss_df = pd.DataFrame(ai_brain.history.history)
```

```
loss_df.plot()
```





```
scaler = StandardScaler()
scaler.fit(X_train)
```

```
StandardScaler
StandardScaler()
```

```
X_test1 = scaler.transform(X_test)
```



```
ai_brain.evaluate(X_test1,y_test)
```

 1/1  0s 94ms/step - loss: 6373.4326  
6373.4326171875

```
X_n1 = [[50]]
```

```
X_n1_1 = scaler.transform(X_n1)
```

```
ai_brain.predict(X_n1_1)
```

 1/1  0s 46ms/step  
array([[235.92375]], dtype=float32)