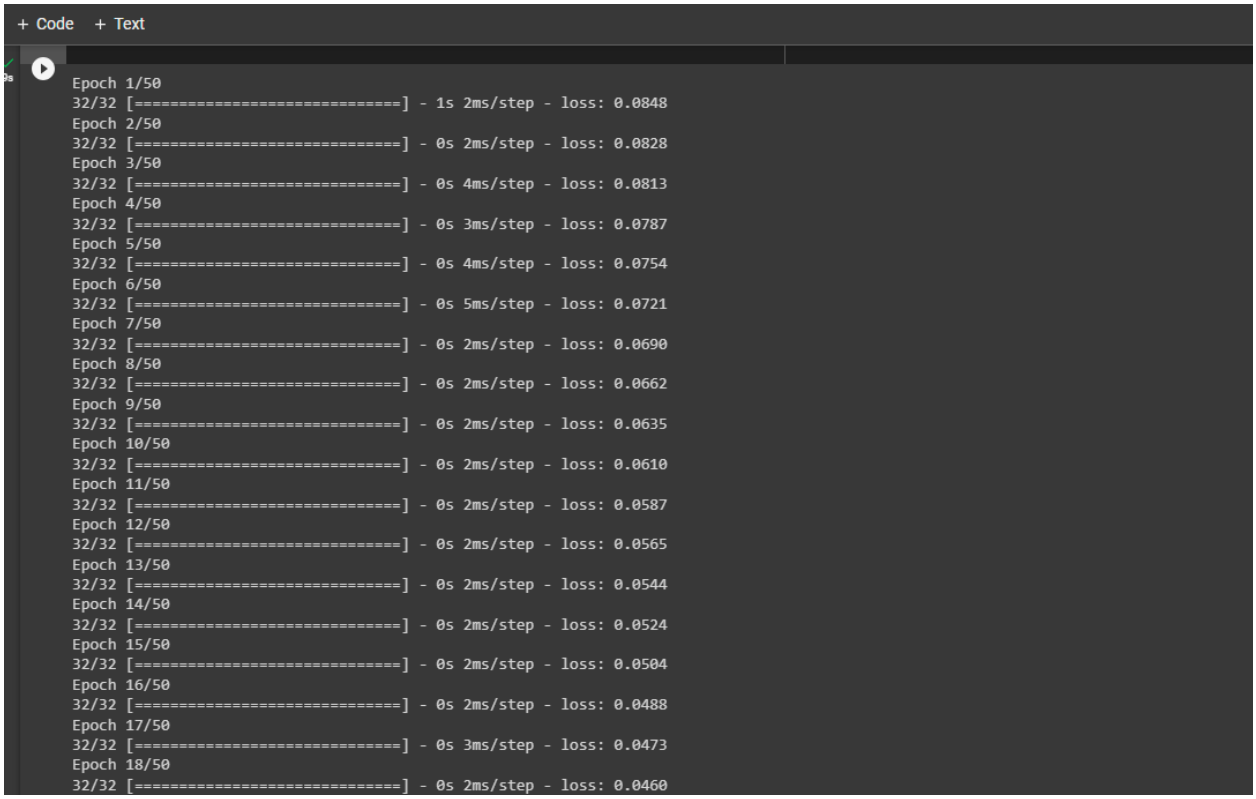
# Experiment no: 5

**Name:Vaishnavi Saindane**
**Roll no: 74**

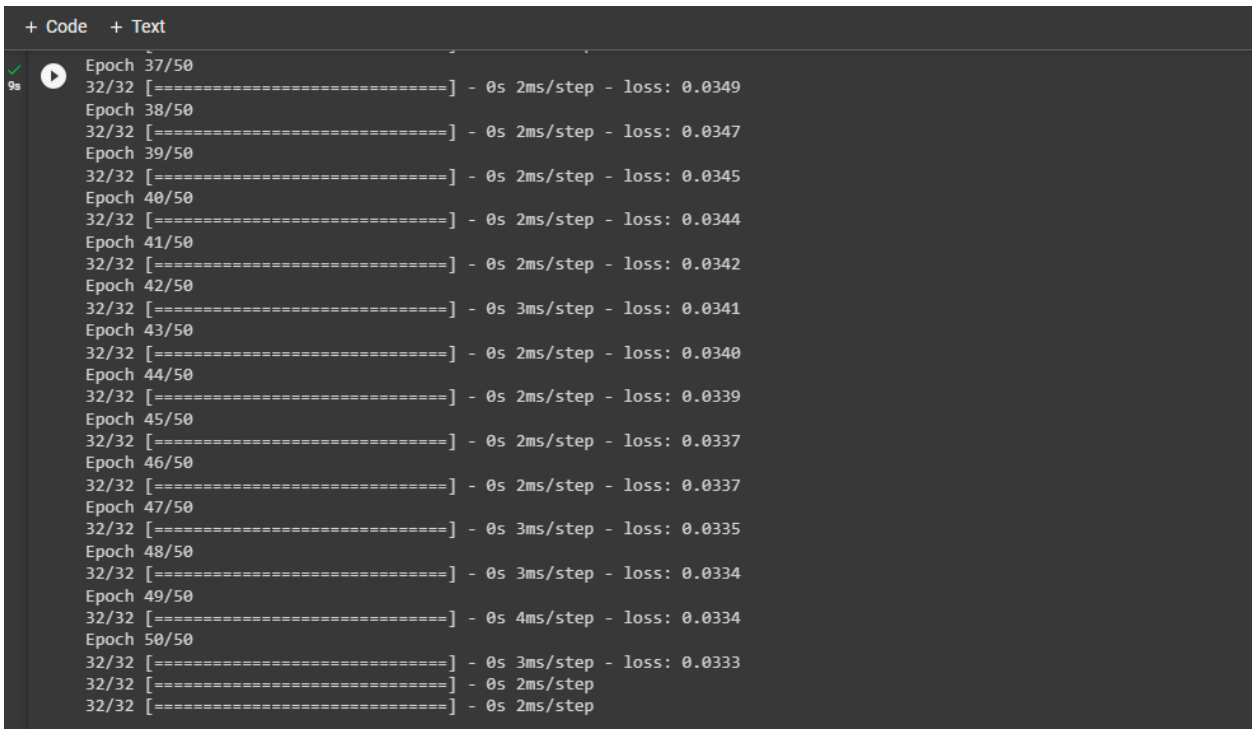
**Code:**

```
import numpy as np
import tensorflow as tf
from tensorflow.keras.layers import Input, Dense
from tensorflow.keras.models import Model

# Generate some sample data
# Here, we'll generate random data with 100 features
num_samples = 1000
num_features = 100
data = np.random.rand(num_samples, num_features)

# Define the autoencoder architecture
input_layer = Input(shape=(num_features,))
encoded = Dense(64, activation='relu')(input_layer)  # Encoding layer with 64 neurons
decoded = Dense(num_features, activation='sigmoid')(encoded)  # Decoding layer

# Create the autoencoder model
autoencoder = Model(inputs=input_layer, outputs=decoded)

# Compile the model
autoencoder.compile(optimizer='adam', loss='mean_squared_error')

# Train the autoencoder
num_epochs = 50
batch_size = 32

autoencoder.fit(data, data, epochs=num_epochs, batch_size=batch_size)

# After training, you can use the encoder part of the model to obtain encoded representations
encoder = Model(inputs=input_layer, outputs=encoded)
encoded_data = encoder.predict(data)

# You can also use the autoencoder for generating reconstructed data
reconstructed_data = autoencoder.predict(data)
```

## Output:



```
+ Code + Text

Epoch 1/50
32/32 [==============================] - 1s 2ms/step - loss: 0.0848
Epoch 2/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0828
Epoch 3/50
32/32 [==============================] - 0s 4ms/step - loss: 0.0813
Epoch 4/50
32/32 [==============================] - 0s 3ms/step - loss: 0.0787
Epoch 5/50
32/32 [==============================] - 0s 4ms/step - loss: 0.0754
Epoch 6/50
32/32 [==============================] - 0s 5ms/step - loss: 0.0721
Epoch 7/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0690
Epoch 8/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0662
Epoch 9/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0635
Epoch 10/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0610
Epoch 11/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0587
Epoch 12/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0565
Epoch 13/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0544
Epoch 14/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0524
Epoch 15/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0504
Epoch 16/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0488
Epoch 17/50
32/32 [==============================] - 0s 3ms/step - loss: 0.0473
Epoch 18/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0460
```



```
+ Code + Text

Epoch 37/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0349
Epoch 38/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0347
Epoch 39/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0345
Epoch 40/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0344
Epoch 41/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0342
Epoch 42/50
32/32 [==============================] - 0s 3ms/step - loss: 0.0341
Epoch 43/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0340
Epoch 44/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0339
Epoch 45/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0337
Epoch 46/50
32/32 [==============================] - 0s 2ms/step - loss: 0.0337
Epoch 47/50
32/32 [==============================] - 0s 3ms/step - loss: 0.0335
Epoch 48/50
32/32 [==============================] - 0s 3ms/step - loss: 0.0334
Epoch 49/50
32/32 [==============================] - 0s 4ms/step - loss: 0.0334
Epoch 50/50
32/32 [==============================] - 0s 3ms/step - loss: 0.0333
32/32 [==============================] - 0s 2ms/step
32/32 [==============================] - 0s 2ms/step
```