```
import numpy as np
import matplotlib.pyplot as plt
from tensorflow.keras.layers import Input, Dense
from tensorflow.keras.models import Model
from tensorflow.keras.datasets import mnist
# Load the MNIST dataset
(x_train, _), (x_test, _) = mnist.load_data()
# Normalize the data to the range of [0, 1]
x_train = x_train.astype('float32') / 255.0
x_{\text{test}} = x_{\text{test.astype}}(\text{'float32'}) / 255.0
# Flatten the 28x28 images into vectors of size 784
x train = x train.reshape((x train.shape[0], -1))
x_test = x_test.reshape((x_test.shape[0], -1))
# Define the dimensions of the autoencoder
input\_dim = x\_train.shape[1] # 784
encoding_dim = 32  # Dimension for the latent space
# Define the autoencoder
input_layer = Input(shape=(input_dim,))
encoded = Dense(encoding_dim, activation='relu')(input_layer)
decoded = Dense(input_dim, activation='sigmoid')(encoded)
# Build the autoencoder model
autoencoder = Model(inputs=input_layer, outputs=decoded)
# Build the encoder model
encoder = Model(inputs=input_layer, outputs=encoded)
# Compile the autoencoder
autoencoder.compile(optimizer='adam', loss='binary_crossentropy')
# Train the autoencoder
autoencoder.fit(x_train, x_train,
                epochs=50,
                batch_size=256,
                shuffle=True,
                validation_data=(x_test, x_test))
# Use the encoder to transform the test data
encoded_data = encoder.predict(x_test)
# Visualize some original and reconstructed images
decoded_images = autoencoder.predict(x_test)
n = 10 # Number of images to display
plt.figure(figsize=(10, 4))
for i in range(n):
    # Display original images
    ax = plt.subplot(2, n, i + 1)
    plt.imshow(x_test[i].reshape(28, 28), cmap='gray')
    plt.title("Original")
    plt.axis('off')
    # Display reconstructed images
    ax = plt.subplot(2, n, i + 1 + n)
    plt.imshow(decoded_images[i].reshape(28, 28), cmap='gray')
    plt.title("Reconstructed")
    plt.axis('off')
plt.show()
# Print the dimensions of the encoded data
print("Encoded data shape:", encoded_data.shape)
```

_			
$\rightarrow$	Epoch 1/50		
		3s	10ms/step - loss: 0.3867 - val_loss: 0.1921
	Epoch 2/50	_	
		35	13ms/step - loss: 0.1806 - val_loss: 0.1536
	Epoch 3/50	2-	Ome/stern least 0 1400 well-least 0 1337
	235/235 — Epoch 4/50	25	9ms/step - loss: 0.1490 - val_loss: 0.1337
		2 c	9ms/step - loss: 0.1314 - val_loss: 0.1208
	Epoch 5/50	23	Уш3/ 3 сер - 1033. 0.1314 - Val_1033. 0.1200
	•	2s	9ms/step - loss: 0.1200 - val_loss: 0.1128
	Epoch 6/50		
	·	3s	9ms/step - loss: 0.1127 - val_loss: 0.1068
	Epoch 7/50		
	235/235	3s	13ms/step - loss: 0.1072 - val_loss: 0.1027
	Epoch 8/50		
		25	9ms/step - loss: 0.1032 - val_loss: 0.0998
	Epoch 9/50 235/235 ————————————————————————————————————	3 c	9ms/step - loss: 0.1003 - val_loss: 0.0975
	Epoch 10/50		
	•	2s	9ms/step - loss: 0.0982 - val_loss: 0.0960
	Epoch 11/50		
		3s	9ms/step - loss: 0.0969 - val_loss: 0.0949
	Epoch 12/50	2.	12ms/stan loss: 0.0061 val loss: 0.0041
	<b>235/235</b> Epoch 13/50	25	13ms/step - loss: 0.0961 - val_loss: 0.0941
	•	4s	9ms/step - loss: 0.0953 - val_loss: 0.0936
	Epoch 14/50		.,,
	235/235 ————	3s	9ms/step - loss: 0.0949 - val_loss: 0.0933
	Epoch 15/50		
		3s	9ms/step - loss: 0.0945 - val_loss: 0.0930
	Epoch 16/50	2.	12ms/step - loss: 0.0942 - val loss: 0.0928
	<b>235/235</b> — Epoch 17/50	25	12ms/step - 10ss. 0.0942 - Val_10ss. 0.0928
	·	3s	11ms/step - loss: 0.0941 - val_loss: 0.0926
	Epoch 18/50		<u>-</u>
	235/235	2s	9ms/step - loss: 0.0939 - val_loss: 0.0925
	Epoch 19/50	_	
		35	9ms/step - loss: 0.0937 - val_loss: 0.0924
	Epoch 20/50 235/235 ————————————————————————————————————	35	9ms/step - loss: 0.0936 - val_loss: 0.0924
	Epoch 21/50	-	Ушу, эсср 1033. 0.0930 Var_1033. 0.0924
	235/235	3s	11ms/step - loss: 0.0935 - val_loss: 0.0922
	Epoch 22/50		
	235/235	3s	13ms/step - loss: 0.0934 - val_loss: 0.0921
	Epoch 23/50	2-	0/
	<b>235/235</b> — Epoch 24/50	25	9ms/step - loss: 0.0934 - val_loss: 0.0921
		35	9ms/step - loss: 0.0933 - val loss: 0.0920
	Epoch 25/50		, <u>-</u>
	235/235 —	2s	9ms/step - loss: 0.0934 - val_loss: 0.0920
	Epoch 26/50		
		3s	9ms/step - loss: 0.0933 - val_loss: 0.0920
	Epoch 27/50	4.5	14ms/step - loss: 0.0933 - val_loss: 0.0919
	<b>235/235</b> — Epoch 28/50	45	141113/3CEP - 1033. 0.0933 - Val_1033. 0.0919
	•	2s	9ms/step - loss: 0.0930 - val loss: 0.0919
	Epoch 29/50		
		2s	9ms/step - loss: 0.0934 - val_loss: 0.0920
	Epoch 30/50	_	
		3s	9ms/step - loss: 0.0930 - val_loss: 0.0918
	Epoch 31/50 235/235	2.	9ms/step - loss: 0.0929 - val_loss: 0.0918
	Epoch 32/50	,,	лыз/ эсер 1033. 0.0929 var_1033. 0.0910
		3s	13ms/step - loss: 0.0929 - val_loss: 0.0918
	Epoch 33/50		
	235/235 —	4s	9ms/step - loss: 0.0927 - val_loss: 0.0918
	Epoch 34/50	_	
	<b>235/235</b> Epoch 35/50	25	9ms/step - loss: 0.0930 - val_loss: 0.0917
	•	25	9ms/step - loss: 0.0929 - val_loss: 0.0918
	Epoch 36/50		Jans, 300p 1033. 0.0323 Val_1033. 0.0310
	•	3s	11ms/step - loss: 0.0930 - val_loss: 0.0917
	Epoch 37/50		
		3s	12ms/step - loss: 0.0929 - val_loss: 0.0917
	Epoch 38/50 235/235 ————————————————————————————————————	25	9ms/step - loss: 0.0927 - val_loss: 0.0917
	Epoch 39/50	25	- val_1055. 0.091/
	•	3s	8ms/step - loss: 0.0928 - val_loss: 0.0917
	Epoch 40/50	_	
		2s	9ms/step - loss: 0.0929 - val_loss: 0.0916
	Epoch 41/50	_	0/
		25	9ms/step - loss: 0.0927 - val_loss: 0.0917
	Epoch 42/50 235/235	3 <	13ms/step - loss: 0.0926 - val_loss: 0.0917
	E 1 43 /FO		

Epoch 43/50										
235/235 —		2s 10m	s/step - l	oss: 0.092	6 - val_lo	ss: 0.0916				
Epoch 44/50										
235/235		—— <b>2s</b> 9ms	/step - lo	ss: 0.0927	- val_los	s: 0.0915				
Epoch 45/50										
235/235		—— <b>2s</b> 9ms	/step - lo	ss: 0.0927	- val_los	s: 0.0916				
Epoch 46/50										
235/235 ———		3s 9ms	/step - lo	ss: 0.092/	- val_los	s: 0.0916				
Epoch 47/50		2- 11-	-/- <del>-</del> 1	0 003	0	0 0016				
235/235		3S 11m	s/step - 1	055: 0.092	8 - val_lo	55: 0.0916				
Epoch 48/50		Fs 0ms	/ston lo	0 0023	- val los	c. 0 0016				
<b>235/235</b> Epoch 49/50		<b>35</b> 91115	/step = 10	55. 0.0923	- Val_105	5. 0.0910				
235/235		<b>2s</b> 9ms	/ston = lo	cc. 0 0027	- val loss	c · 0 0016				
Epoch 50/50		23 91113	/scep - 10	33. 0.0327	- vai_103	3. 0.0910				
235/235		3s 9ms	/sten = lo	55. 0.0925	- val los	s · 0 .0916				
313/313 —		- <b>3s</b> 9ms/step - loss: 0.0925 - val_loss: 0.0916 - <b>0s</b> 1ms/step								
313/313 —		—— <b>1s</b> 2ms								
•	0::-			0 : : 1	0 : : - 1	0 : : 1	0 : : 1	0::-		
Original	Original	Original	Original	Original	Original	Original	Original	Original	(	
					10.20	Obst. 1000		-		



 $ReconstruR \underline{\textbf{w}} \underline{\textbf{e}} \underline{\textbf{d}} \underline{\textbf{n}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{w}} \underline{\textbf{e}} \underline{\textbf{d}} \underline{\textbf{n}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{w}} \underline{\textbf{w}} \underline{\textbf{e}} \underline{\textbf{d}} \underline{\textbf{n}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{w}} \underline{\textbf{w}} \underline{\textbf{e}} \underline{\textbf{d}} \underline{\textbf{n}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{s}} \underline{\textbf{s}} \underline{\textbf{r}} \underline{\textbf{s}} \underline{\textbf{s}}} \underline{\textbf{s}} \underline{\textbf{s}}}$ 



Encoded data shape: (10000, 32)