Loading MNIST

In [10]:
```python
from keras.datasets import mnist
(x_train,_),(x_test,_)=mnist.load_data()
```

Checking Dimensions

In [11]:
```python
print(x_train.shape)
```

```
(60000, 28, 28)
```

Image Pre-processing

In [12]:
```python
def pre_process(X):
    X = X/255.0
    X = X.reshape((len(X), 784))
    return X

x_train  =  pre_process(x_train)
x_test  =  pre_process(x_test)

print("X_train", x_train.shape)
print("X_test", x_test.shape)
```

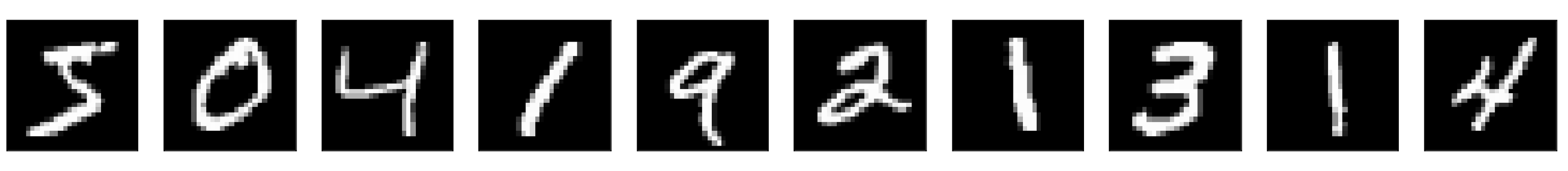
```
X_train (60000, 784)
X_test (10000, 784)
```

Visualized the training images

In [13]:
```python
import matplotlib.pyplot as plt
def show_data(X, n=10, height=28, width=28, title=""):
    plt.figure(figsize=(20, 5))
    for i in range(n):
        ax = plt.subplot(2,n,i+1)
        plt.imshow(X[i].reshape((height,width)))
        plt.gray()
        ax.get_xaxis().set_visible(False)
        ax.get_yaxis().set_visible(False)
    plt.suptitle(title, fontsize = 20)
```

In [14]:
```python
show_data(x_train,title="Training images")
```

Training images



Building the Stacked Autoencoder

- Input layer = 784 neurons
- Hidden layer_1 = 256 neurons
- Code layer = 48 neurons
- Hidden layer_2 = 256 neurons
- Output layer = 784 neurons

In [15]:
```python
from keras.layers import Input, Dense
from keras.models import Model
```

In [16]:
```python
# encoder
input_layer = Input(shape=(784,), name="INPUT")
hidden_layer_1 = Dense(256, activation='relu', name="HIDDEN_1")(input_layer)


# code
code_layer = Dense(100, activation='relu', name="CODE")(hidden_layer_1)

# decoder

hidden_layer_2 = Dense(256, activation='relu', name="HIDDEN_4")(code_layer)
output_layer = Dense(784, activation='sigmoid', name="OUTPUT")(hidden_layer_2)
```

Compiling the layers

In [17]:
```python
stacked_autoencoder=Model(input_layer, output_layer)
stacked_autoencoder.compile(optimizer="Adam",loss="binary_crossentropy")
stacked_autoencoder.summary()
```

```
Model: "model_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
INPUT (InputLayer)           [(None, 784)]             0
_________________________________________________________________
HIDDEN_1 (Dense)             (None, 256)               200960
_________________________________________________________________
CODE (Dense)                 (None, 100)               25700
_________________________________________________________________
HIDDEN_4 (Dense)             (None, 256)               25856
_________________________________________________________________
OUTPUT (Dense)               (None, 784)               201488
=================================================================
Total params: 454,004
Trainable params: 454,004
Non-trainable params: 0
_________________________________________________________________
```

Fit into the neural network

In [18]:
```python
stacked_autoencoder.fit(x_train, x_train, epochs=100, batch_size=256,shuffle=True,
                        validation_data=(x_test, x_test))
```

```
Epoch 1/100
235/235 [==============================] - 8s 31ms/step - loss: 0.2913 - val_loss: 0.1149
Epoch 2/100
235/235 [==============================] - 7s 28ms/step - loss: 0.1096 - val_loss: 0.0946
Epoch 3/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0938 - val_loss: 0.0871
Epoch 4/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0870 - val_loss: 0.0828
Epoch 5/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0834 - val_loss: 0.0805
Epoch 6/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0805 - val_loss: 0.0781
Epoch 7/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0786 - val_loss: 0.0769
Epoch 8/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0773 - val_loss: 0.0758
Epoch 9/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0761 - val_loss: 0.0749
Epoch 10/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0753 - val_loss: 0.0742
Epoch 11/100
235/235 [==============================] - 8s 32ms/step - loss: 0.0744 - val_loss: 0.0733
Epoch 12/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0737 - val_loss: 0.0728
Epoch 13/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0731 - val_loss: 0.0723
Epoch 14/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0726 - val_loss: 0.0718
Epoch 15/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0722 - val_loss: 0.0713
Epoch 16/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0718 - val_loss: 0.0712
Epoch 17/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0714 - val_loss: 0.0709
Epoch 18/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0710 - val_loss: 0.0705
Epoch 19/100
235/235 [==============================] - 7s 31ms/step - loss: 0.0710 - val_loss: 0.0704
Epoch 20/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0704 - val_loss: 0.0700
Epoch 21/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0703 - val_loss: 0.0700
Epoch 22/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0701 - val_loss: 0.0696
Epoch 23/100
235/235 [==============================] - 8s 32ms/step - loss: 0.0698 - val_loss: 0.0694
Epoch 24/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0696 - val_loss: 0.0699
Epoch 25/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0694 - val_loss: 0.0691
Epoch 26/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0693 - val_loss: 0.0690
Epoch 27/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0693 - val_loss: 0.0689
Epoch 28/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0691 - val_loss: 0.0687
Epoch 29/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0690 - val_loss: 0.0686
Epoch 30/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0688 - val_loss: 0.0685
Epoch 31/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0686 - val_loss: 0.0685
Epoch 32/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0686 - val_loss: 0.0683
Epoch 33/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0684 - val_loss: 0.0682
Epoch 34/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0683 - val_loss: 0.0682
Epoch 35/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0682 - val_loss: 0.0680
Epoch 36/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0682 - val_loss: 0.0681
Epoch 37/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0682 - val_loss: 0.0678
Epoch 38/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0680 - val_loss: 0.0680
Epoch 39/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0680 - val_loss: 0.0679
Epoch 40/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0679 - val_loss: 0.0677
Epoch 41/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0678 - val_loss: 0.0676
Epoch 42/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0677 - val_loss: 0.0678
Epoch 43/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0676 - val_loss: 0.0676
Epoch 44/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0677 - val_loss: 0.0675
Epoch 45/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0676 - val_loss: 0.0674
Epoch 46/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0675 - val_loss: 0.0675
Epoch 47/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0676 - val_loss: 0.0675
Epoch 48/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0674 - val_loss: 0.0673
Epoch 49/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0675 - val_loss: 0.0672
Epoch 50/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0672 - val_loss: 0.0673
Epoch 51/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0673 - val_loss: 0.0673
Epoch 52/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0672 - val_loss: 0.0673
Epoch 53/100
235/235 [==============================] - 7s 32ms/step - loss: 0.0673 - val_loss: 0.0671
Epoch 54/100
235/235 [==============================] - 7s 32ms/step - loss: 0.0671 - val_loss: 0.0671
Epoch 55/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0671 - val_loss: 0.0673
Epoch 56/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0672 - val_loss: 0.0673
Epoch 57/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0673 - val_loss: 0.0670
Epoch 58/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0670 - val_loss: 0.0670
Epoch 59/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0671 - val_loss: 0.0671
Epoch 60/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0671 - val_loss: 0.0670
Epoch 61/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0671 - val_loss: 0.0670
Epoch 62/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0670 - val_loss: 0.0671
Epoch 63/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0670 - val_loss: 0.0672
Epoch 64/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0671 - val_loss: 0.0669
Epoch 65/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0668 - val_loss: 0.0669
Epoch 66/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0669 - val_loss: 0.0668
Epoch 67/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0668 - val_loss: 0.0670
Epoch 68/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0670 - val_loss: 0.0669
Epoch 69/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0668 - val_loss: 0.0670
Epoch 70/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0670 - val_loss: 0.0669
Epoch 71/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0668 - val_loss: 0.0669
Epoch 72/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0667 - val_loss: 0.0667
Epoch 73/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0668 - val_loss: 0.0670
Epoch 74/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0669 - val_loss: 0.0668
Epoch 75/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0668 - val_loss: 0.0667
Epoch 76/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0666 - val_loss: 0.0667
Epoch 77/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0668 - val_loss: 0.0668
Epoch 78/100
235/235 [==============================] - 6s 26ms/step - loss: 0.0668 - val_loss: 0.0668
Epoch 79/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0668 - val_loss: 0.0668
Epoch 80/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0668 - val_loss: 0.0667
Epoch 81/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0665 - val_loss: 0.0670
Epoch 82/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0666 - val_loss: 0.0668
Epoch 83/100
235/235 [==============================] - 6s 25ms/step - loss: 0.0667 - val_loss: 0.0669
Epoch 84/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0667 - val_loss: 0.0667
Epoch 85/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0668 - val_loss: 0.0668
Epoch 86/100
235/235 [==============================] - 7s 30ms/step - loss: 0.0666 - val_loss: 0.0668
Epoch 87/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0667 - val_loss: 0.0667
Epoch 88/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0666 - val_loss: 0.0666
Epoch 89/100
235/235 [==============================] - 6s 27ms/step - loss: 0.0667 - val_loss: 0.0667
Epoch 90/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0666 - val_loss: 0.0667
Epoch 91/100
235/235 [==============================] - 8s 32ms/step - loss: 0.0667 - val_loss: 0.0666
Epoch 92/100
235/235 [==============================] - 8s 36ms/step - loss: 0.0666 - val_loss: 0.0666
Epoch 93/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0666 - val_loss: 0.0666
Epoch 94/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0666 - val_loss: 0.0666
Epoch 95/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0666 - val_loss: 0.0666
Epoch 96/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0667 - val_loss: 0.0669
Epoch 97/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0668 - val_loss: 0.0668
Epoch 98/100
235/235 [==============================] - 7s 28ms/step - loss: 0.0666 - val_loss: 0.0666
Epoch 99/100
235/235 [==============================] - 8s 33ms/step - loss: 0.0665 - val_loss: 0.0667
Epoch 100/100
235/235 [==============================] - 7s 29ms/step - loss: 0.0666 - val_loss: 0.0665
```

Out[18]:
```
<tensorflow.python.keras.callbacks.History at 0x7fe40df064f0>
```

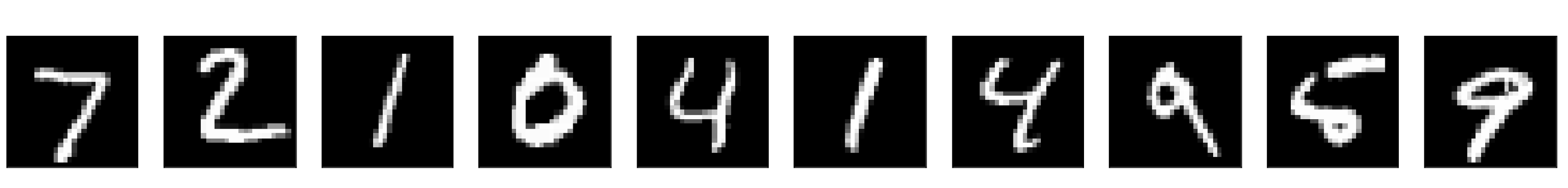
Predicting

In [19]:
```python
decoded_data = stacked_autoencoder.predict(x_test)
```

Visualization

In [20]:
```python
show_data(x_test, title="original data")
show_data(decoded_data, title="decoded data")
```

original data



decoded data

In [ ]: