In [ ]:
```
# Assignment no 8
import numpy as np
import tensorflow as tf
from tensorflow.keras.layers import Input, Dense
from tensorflow.keras.models import Model
from tensorflow.keras.datasets import mnist
```

In [ ]:
```
(X_train,y_train),(X_test,y_test)=mnist.load_data()
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mni
st.npz
11490434/11490434 [==============================] - 0s 0us/step
```

In [ ]:
```
X_test =X_test.astype('float32')/255.0
X_train =X_train.astype('float32')/255.0
```

In [ ]:
```
X_train = X_train.reshape((len(X_train), np.prod(X_train.shape[1:])))
X_test = X_test.reshape((len(X_test), np.prod(X_test.shape[1:])))
```

In [ ]:
```
noise_factor = 0.5
x_train_noisy = X_train + noise_factor * np.random.normal(loc=0.0, scale=1.0, size=X_t
x_test_noisy = X_test + noise_factor * np.random.normal(loc=0.0, scale=1.0, size=X_tes

x_train_noisy = np.clip(x_train_noisy, 0., 1.)
x_test_noisy = np.clip(x_test_noisy, 0., 1.)
```

In [ ]:
```
input_img = Input(shape=(784,))
encoded = Dense(128, activation='relu')(input_img)
decoded = Dense(784, activation='sigmoid')(encoded)

autoencoder = Model(input_img, decoded)
```

In [ ]:
```
autoencoder.compile(optimizer='adam', loss='binary_crossentropy')
```

In [ ]:
```
autoencoder.fit(x_train_noisy, X_train, epochs=50, batch_size=256, shuffle=True, valid
```

```
Epoch 1/50
235/235 [==============================] - 5s 19ms/step - loss: 0.2373 - val_loss: 0.
1708
Epoch 2/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1547 - val_loss: 0.
1401
Epoch 3/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1343 - val_loss: 0.
1274
Epoch 4/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1251 - val_loss: 0.
1212
Epoch 5/50
235/235 [==============================] - 5s 20ms/step - loss: 0.1200 - val_loss: 0.
1179
Epoch 6/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1172 - val_loss: 0.
1156
Epoch 7/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1153 - val_loss: 0.
1145
Epoch 8/50
235/235 [==============================] - 5s 20ms/step - loss: 0.1139 - val_loss: 0.
1133
Epoch 9/50
235/235 [==============================] - 4s 16ms/step - loss: 0.1128 - val_loss: 0.
1126
Epoch 10/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1119 - val_loss: 0.
1118
Epoch 11/50
235/235 [==============================] - 4s 17ms/step - loss: 0.1111 - val_loss: 0.
1111
Epoch 12/50
235/235 [==============================] - 4s 17ms/step - loss: 0.1104 - val_loss: 0.
1105
Epoch 13/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1097 - val_loss: 0.
1099
Epoch 14/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1092 - val_loss: 0.
1096
Epoch 15/50
235/235 [==============================] - 5s 20ms/step - loss: 0.1088 - val_loss: 0.
1093
Epoch 16/50
235/235 [==============================] - 4s 16ms/step - loss: 0.1083 - val_loss: 0.
1088
Epoch 17/50
235/235 [==============================] - 7s 31ms/step - loss: 0.1080 - val_loss: 0.
1085
Epoch 18/50
235/235 [==============================] - 7s 29ms/step - loss: 0.1075 - val_loss: 0.
1082
Epoch 19/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1072 - val_loss: 0.
1079
Epoch 20/50
235/235 [==============================] - 5s 20ms/step - loss: 0.1069 - val_loss: 0.
1076
```

```
Epoch 21/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1067 - val_loss: 0.
1075
Epoch 22/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1065 - val_loss: 0.
1076
Epoch 23/50
235/235 [==============================] - 4s 16ms/step - loss: 0.1063 - val_loss: 0.
1072
Epoch 24/50
235/235 [==============================] - 4s 17ms/step - loss: 0.1060 - val_loss: 0.
1072
Epoch 25/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1059 - val_loss: 0.
1071
Epoch 26/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1057 - val_loss: 0.
1068
Epoch 27/50
235/235 [==============================] - 4s 19ms/step - loss: 0.1056 - val_loss: 0.
1068
Epoch 28/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1055 - val_loss: 0.
1066
Epoch 29/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1054 - val_loss: 0.
1065
Epoch 30/50
235/235 [==============================] - 4s 16ms/step - loss: 0.1052 - val_loss: 0.
1065
Epoch 31/50
235/235 [==============================] - 4s 17ms/step - loss: 0.1051 - val_loss: 0.
1064
Epoch 32/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1050 - val_loss: 0.
1064
Epoch 33/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1049 - val_loss: 0.
1062
Epoch 34/50
235/235 [==============================] - 4s 19ms/step - loss: 0.1048 - val_loss: 0.
1063
Epoch 35/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1048 - val_loss: 0.
1063
Epoch 36/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1047 - val_loss: 0.
1063
Epoch 37/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1046 - val_loss: 0.
1061
Epoch 38/50
235/235 [==============================] - 4s 18ms/step - loss: 0.1046 - val_loss: 0.
1061
Epoch 39/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1045 - val_loss: 0.
1061
Epoch 40/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1045 - val_loss: 0.
1062
```

```
Epoch 41/50
235/235 [==============================] - 4s 18ms/step - loss: 0.1045 - val_loss: 0.
1062
Epoch 42/50
235/235 [==============================] - 4s 16ms/step - loss: 0.1045 - val_loss: 0.
1061
Epoch 43/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1044 - val_loss: 0.
1060
Epoch 44/50
235/235 [==============================] - 3s 14ms/step - loss: 0.1044 - val_loss: 0.
1061
Epoch 45/50
235/235 [==============================] - 5s 21ms/step - loss: 0.1043 - val_loss: 0.
1060
Epoch 46/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1043 - val_loss: 0.
1060
Epoch 47/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1043 - val_loss: 0.
1060
Epoch 48/50
235/235 [==============================] - 5s 20ms/step - loss: 0.1042 - val_loss: 0.
1061
Epoch 49/50
235/235 [==============================] - 3s 15ms/step - loss: 0.1042 - val_loss: 0.
1060
Epoch 50/50
235/235 [==============================] - 4s 17ms/step - loss: 0.1042 - val_loss: 0.
1060
```

Out[ ]:
```
<keras.src.callbacks.History at 0x79dab72d6680>
```

In [ ]:
```
encoder = Model(input_img, encoded)
```

In [ ]:
```
denoised_images = autoencoder.predict(x_test_noisy)
```

```
313/313 [==============================] - 1s 2ms/step
```

In [ ]:
```
import matplotlib.pyplot as plt

n = 10
plt.figure(figsize=(20, 4))
for i in range(n):

    ax = plt.subplot(3, n, i + 1)
    plt.imshow(X_test[i].reshape(28, 28))
    plt.gray()
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)


    ax = plt.subplot(3, n, i + 1 + n)
    plt.imshow(x_test_noisy[i].reshape(28, 28))
    plt.gray()
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)


    ax = plt.subplot(3, n, i + 1 + 2 * n)
```

```
    plt.imshow(denoised_images[i].reshape(28, 28))
    plt.gray()
    ax.get_xaxis().set_visible(False)
    ax.get_yaxis().set_visible(False)

plt.show()
```