In [1]:
```
import pandas as pd # Import Pandas for data manipulation using dataframes
import numpy as np # Import Numpy for data statistical analysis
import matplotlib.pyplot as plt # Import matplotlib for data visualisation
import seaborn as sns
import random

%matplotlib inline
sns.set_style("whitegrid")
```

In [2]:
```
fashion_train_df = pd.read_csv("C:/Users/hp/Documents/LP5/deep learning/fashio
fashion_test_df = pd.read_csv("C:/Users/hp/Documents/LP5/deep learning/fashion
```

In [3]:
```
fashion_train_df.head()
```

Out[3]:

	label	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel775	pixel77
0	2	0	0	0	0	0	0	0	0	0	...	0	
1	9	0	0	0	0	0	0	0	0	0	...	0	
2	6	0	0	0	0	0	0	0	5	0	...	0	
3	0	0	0	0	1	2	0	0	0	0	...	3	
4	3	0	0	0	0	0	0	0	0	0	...	0	

5 rows × 785 columns

In [4]:
```
fashion_train_df.tail()
```

Out[4]:

	label	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel775	pix
59995	9	0	0	0	0	0	0	0	0	0	...	0	
59996	1	0	0	0	0	0	0	0	0	0	...	73	
59997	8	0	0	0	0	0	0	0	0	0	...	160	
59998	8	0	0	0	0	0	0	0	0	0	...	0	
59999	7	0	0	0	0	0	0	0	0	0	...	0	

5 rows × 785 columns

In [5]:
```
fashion_test_df.head()
```

Out[5]:

	label	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel775	pixel77
0	0	0	0	0	0	0	0	0	9	8	...	103	8
1	1	0	0	0	0	0	0	0	0	0	...	34	
2	2	0	0	0	0	0	0	14	53	99	...	0	
3	2	0	0	0	0	0	0	0	0	0	...	137	12
4	3	0	0	0	0	0	0	0	0	0	...	0	

5 rows × 785 columns

In [6]:
```
fashion_test_df.tail()
```

Out[6]:

	label	pixel1	pixel2	pixel3	pixel4	pixel5	pixel6	pixel7	pixel8	pixel9	...	pixel775	pixe
9995	0	0	0	0	0	0	0	0	0	0	...	32	
9996	6	0	0	0	0	0	0	0	0	0	...	0	
9997	8	0	0	0	0	0	0	0	0	0	...	175	
9998	8	0	1	3	0	0	0	0	0	0	...	0	
9999	1	0	0	0	0	0	0	0	140	119	...	111	

5 rows × 785 columns

In [8]:
```
fashion_train_df.shape
```

Out[8]:
```
(60000, 785)
```

In [9]:
```
train = np.array(fashion_train_df, dtype='float32')
test = np.array(fashion_test_df, dtype='float32')
```

In [10]:
```
train.shape
```

Out[10]:
```
(60000, 785)
```

In [11]:
```
train
```

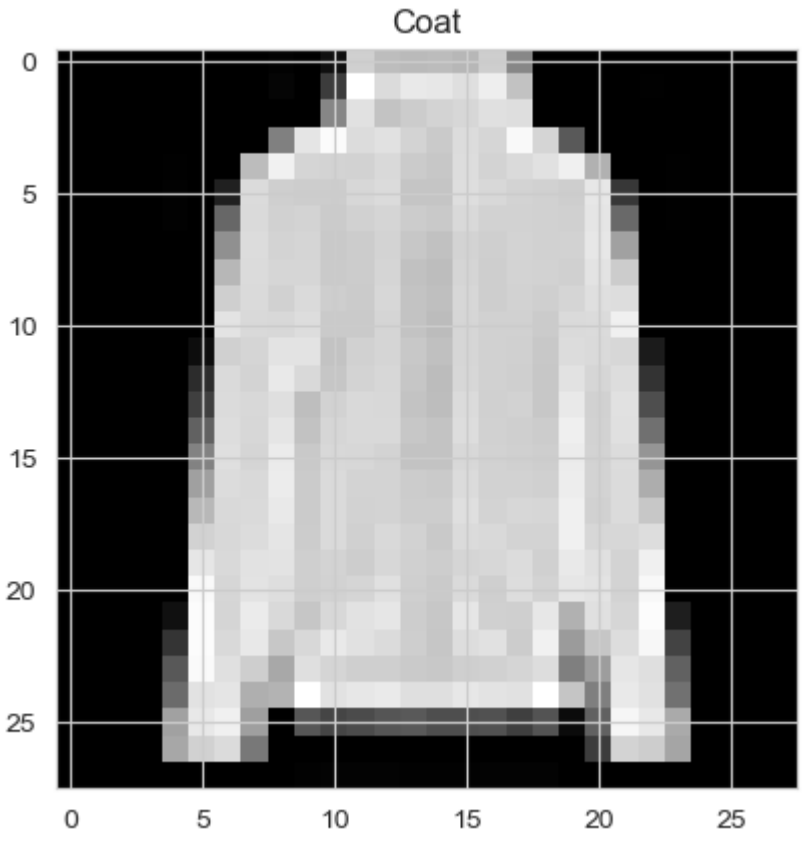
Out[11]:
```
array([[2., 0., 0., ..., 0., 0., 0.],
       [9., 0., 0., ..., 0., 0., 0.],
       [6., 0., 0., ..., 0., 0., 0.],
       ...,
       [8., 0., 0., ..., 0., 0., 0.],
       [8., 0., 0., ..., 0., 0., 0.],
       [7., 0., 0., ..., 0., 0., 0.]], dtype=float32)
```

In [12]:
```
test
```

Out[12]:
```
array([[0., 0., 0., ..., 0., 0., 0.],
       [1., 0., 0., ..., 0., 0., 0.],
       [2., 0., 0., ..., 0., 0., 0.],
       ...,
       [8., 0., 0., ..., 0., 1., 0.],
       [8., 0., 1., ..., 0., 0., 0.],
       [1., 0., 0., ..., 0., 0., 0.]], dtype=float32)
```

In [13]:
```
class_names = ['T_shirt/top', 'Trouser', 'Pullover', 'Dress', 'Coat',
               'Sandal', 'Shirt', 'Sneaker', 'Bag', 'Ankle boot']

# Let's view some images!
i = random.randint(1,60000) # select any random index from 1 to 60,000
plt.imshow(train[i,1:].reshape((28,28))) # reshape and plot the image

plt.imshow(train[i,1:].reshape((28,28)) , cmap = 'gray') # reshape and plot th
label_index = fashion_train_df["label"][i]
plt.title(f"{class_names[label_index]}")
```

Out[13]:
```
Text(0.5, 1.0, 'Coat')
```

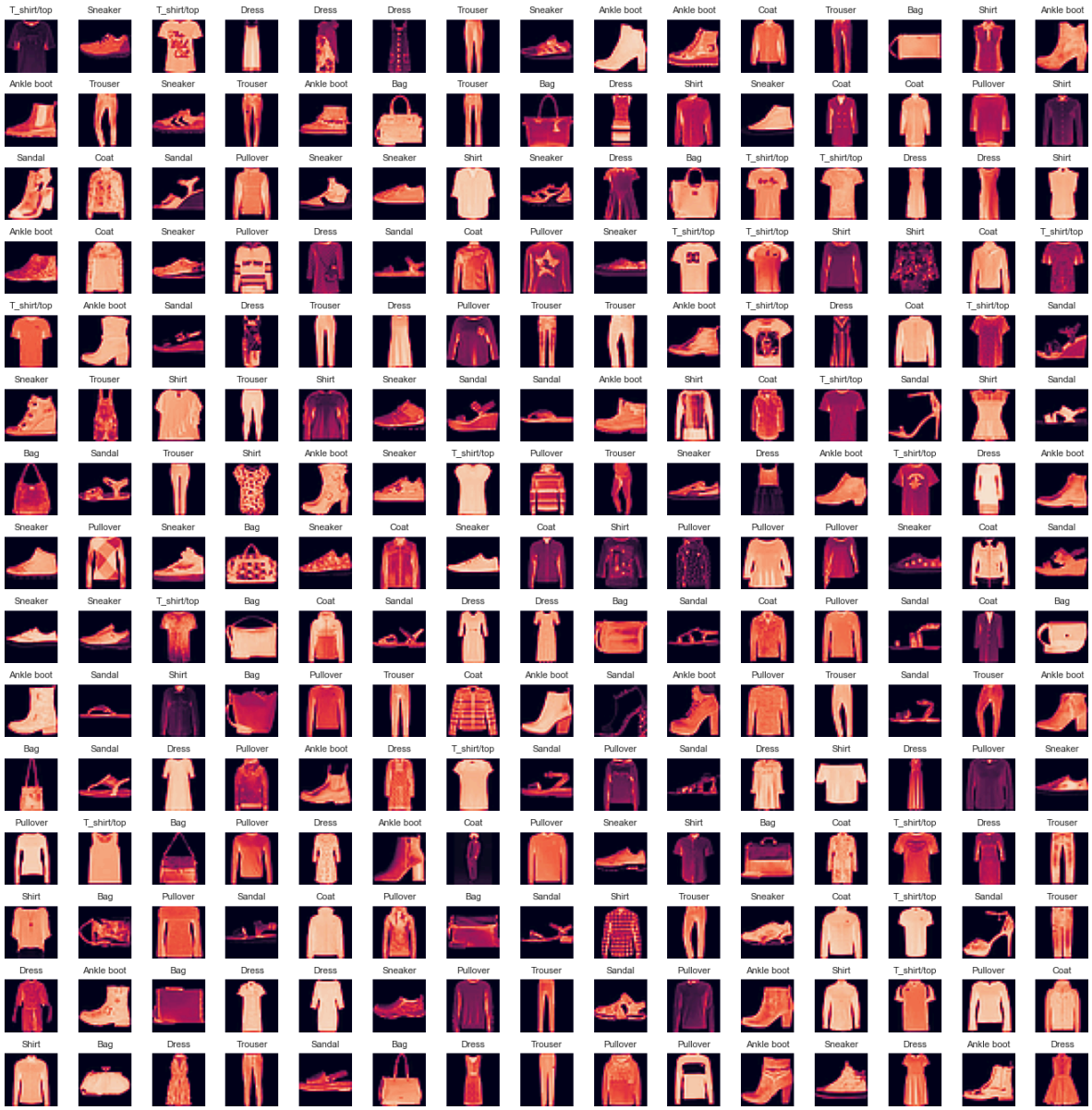


In [14]:
```
label = train[i,0]
label
```

Out[14]:
```
4.0
```

In [15]:

```
W_grid = 15
L_grid = 15

# fig, axes = plt.subplots(L_grid, W_grid)
# subplot return the figure object and axes object
# we can use the axes object to plot specific figures at various locations

fig, axes = plt.subplots(L_grid, W_grid, figsize=(17,17))

axes = axes.ravel() # flaten the 15 x 15 matrix into 225 array

n_train = len(train) # get the length of the train dataset

# Select a random number from 0 to n_train
for i in np.arange(0, W_grid * L_grid): # create evenly spaces variables

    # Select a random number
    index = np.random.randint(0, n_train)
    # read and display an image with the selected index
    axes[i].imshow( train[index,1:].reshape((28,28)) )
    label_index = int(train[index,0])
    axes[i].set_title(class_names[label_index], fontsize=8)
    axes[i].axis('off')

plt.subplots_adjust(hspace=0.4)
```



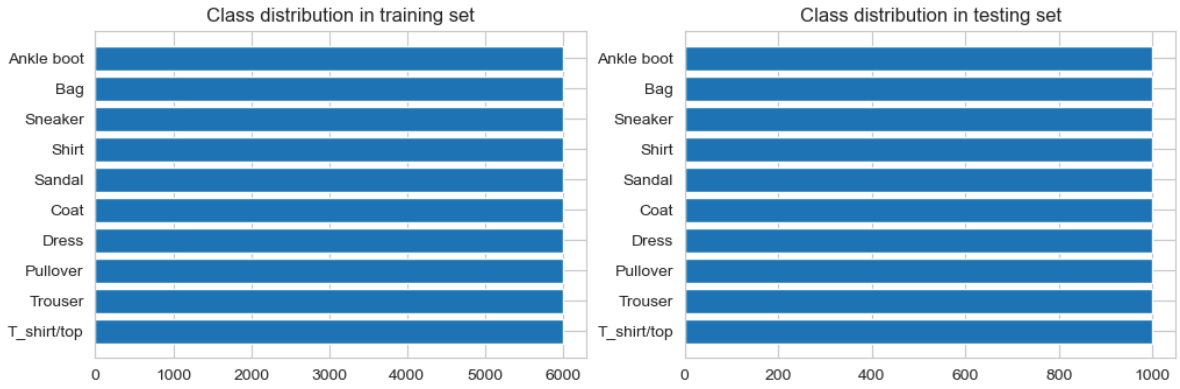
In [16]:
```
X_train = train[:, 1:] / 255
y_train = train[:, 0]

X_test = test[:, 1:] / 255
y_test = test[:,0]
```

In [17]:
```
plt.figure(figsize=(10, 10))
for i in range(25):
    plt.subplot(5, 5, i + 1)
    plt.xticks([])
    plt.yticks([])
    plt.grid(False)
    plt.imshow(X_train[i].reshape((28,28)), cmap=plt.cm.binary)
    label_index = int(y_train[i])
    plt.title(class_names[label_index])
plt.show()
plt.tight_layout()
```


```
<Figure size 640x480 with 0 Axes>
```

In [18]:
```
plt.figure(figsize=(12, 8))

plt.subplot(2, 2, 1)
classes, counts = np.unique(y_train, return_counts=True)
plt.barh(class_names, counts)
plt.title('Class distribution in training set')

plt.subplot(2, 2, 2)
classes, counts = np.unique(y_test, return_counts=True)
plt.barh(class_names, counts)
plt.title('Class distribution in testing set')
```

Out[18]:
```
Text(0.5, 1.0, 'Class distribution in testing set')
```



In [19]:
```
from sklearn.model_selection import train_test_split

X_train, X_validate, y_train, y_validate = train_test_split(X_train, y_train,
```

In [21]:
```
print(X_train.shape)
print(y_train.shape)
```

```
(48000, 784)
(48000,)
```

In [22]:
```
X_train = X_train.reshape(X_train.shape[0], * (28, 28, 1))
X_test = X_test.reshape(X_test.shape[0], * (28, 28, 1))
X_validate = X_validate.reshape(X_validate.shape[0], * (28, 28, 1))
```

In [23]:
```
print(X_train.shape)
print(y_train.shape)
print(X_validate.shape)
print(y_validate.shape)
```

```
(48000, 28, 28, 1)
(48000,)
(12000, 28, 28, 1)
(12000,)
```

In [28]:
```
import keras
import tensorflow as tf
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[28], line 1
----> 1 import keras
      2 import tensorflow as tf

ModuleNotFoundError: No module named 'keras'
```

In [29]:
```
METRICS = [
    'accuracy',
    tf.keras.metrics.Precision(name='precision'),
    tf.keras.metrics.Recall(name='recall')
]

cnn_model.compile(loss ='sparse_categorical_crossentropy', optimizer='adam' ,m
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[29], line 3
      1 METRICS = [
      2     'accuracy',
----> 3     tf.keras.metrics.Precision(name='precision'),
      4     tf.keras.metrics.Recall(name='recall')
      5 ]
      7 cnn_model.compile(loss ='sparse_categorical_crossentropy', optimizer
='adam' ,metrics=['accuracy'])

NameError: name 'tf' is not defined
```

In [26]:
```
predicted_classes = cnn_model.predict(X_test)
predicted_classes = np.argmax(predicted_classes, axis=1
```

```
  Cell In[26], line 2
    predicted_classes = np.argmax(predicted_classes, axis=1
                                                            ^

SyntaxError: unexpected EOF while parsing
```

In [27]:
```
from sklearn.metrics import confusion_matrix
cm = confusion_matrix(y_test, predicted_classes)
plt.figure(figsize = (14,10))
sns.heatmap(cm, annot=True)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[27], line 2
      1 from sklearn.metrics import confusion_matrix
----> 2 cm = confusion_matrix(y_test, predicted_classes)
      3 plt.figure(figsize = (14,10))
      4 sns.heatmap(cm, annot=True)

NameError: name 'predicted_classes' is not defined
```

In [30]:
```
import keras
import tensorflow as tf
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[30], line 1
----> 1 import keras
      2 import tensorflow as tf

ModuleNotFoundError: No module named 'keras'
```

In [31]:
```
import keras


```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[31], line 1
----> 1 import keras

ModuleNotFoundError: No module named 'keras'
```

In [32]:
```
import tensorflow as tf
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
Cell In[32], line 1
----> 1 import tensorflow as tf

ModuleNotFoundError: No module named 'tensorflow'
```

In [ ]:
```

```