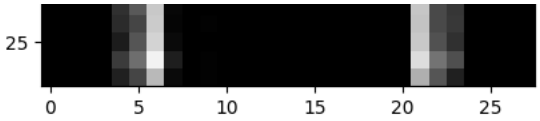
```
import numpy as np
import matplotlib.pyplot as plt
import tensorflow as tf
from tensorflow.keras import layers, models
from tensorflow.keras.datasets import fashion_mnist
from tensorflow.keras.models import Sequential
from tensorflow.keras import layers
from tensorflow.keras.layers import Flatten,Dense

(train_images, train_labels), (test_images, test_labels) = fashion_mnist.load_data()

# Preprocess the data
train_images = train_images/ 255.0
test_images = test_images / 255.0


model = Sequential([
    Flatten(input_shape=(28, 28,1)),
    Dense(128, activation="relu"),
    Dense(128, activation="relu"),
    Dense(64, activation="relu"),
    Dense(10, activation="softmax")
])

class_names = ["T-shirt/top", "Trouser", "Pullover", "Dress", "Coat",
               "Sandal", "Shirt", "Sneaker", "Bag", "Ankle boot"]

model.compile(loss="sparse_categorical_crossentropy", optimizer="adam", metrics=["accuracy"])
history = model.fit(train_images, train_labels, epochs=1, validation_data=(test_images, test_labels))
```

```
1875/1875 [==============================] - 11s 5ms/step - loss: 0.4912 - accuracy: 0.8227 - val_loss: 0.4394 - val_accuracy: 0.84
```

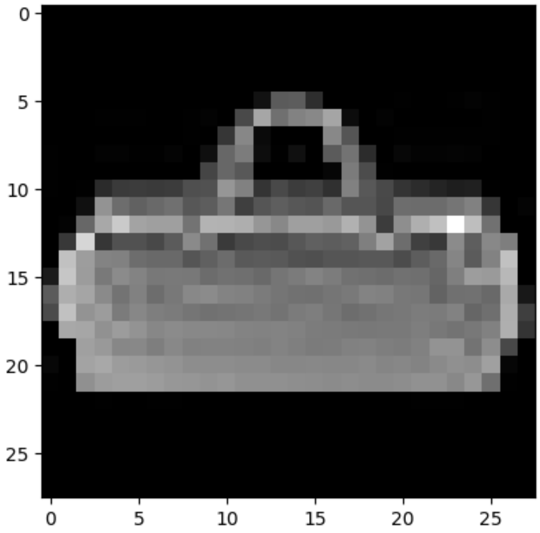
```
loss,accuracy=model.evaluate(test_images,test_labels)
```

```
313/313 [==============================] - 1s 2ms/step - loss: 0.4394 - accuracy: 0.8412
```

```
for _ in range(3):
    index=np.random.randint(0,len(test_images))
    actual_label=test_labels[index]
    predicted_label=np.argmax(model.predict(test_images[index][np.newaxis,...,np.newaxis]))
    print(f'actual_label : {class_names[actual_label]}  Predicted label :{class_names[predicted_label]}')
    plt.imshow(test_images[index],cmap='gray')
    plt.show()
```



```
1/1 [==============================] - 0s 34ms/step
actual_label : Bag  Predicted label :Bag
```



```
1/1 [==============================] - 0s 31ms/step
actual_label : Trouser  Predicted label :Trouser
```