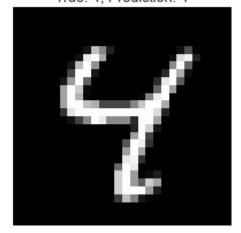
```
In [ ]: import pandas as pd
         import numpy as np
         from sklearn import svm
         from sklearn import metrics
         import matplotlib.pyplot as plt
         import seaborn as sns; sns.set(font_scale=1.2)
         import tensorflow as tf
In [ ]: (X_train, y_train), (X_test, y_test) = tf.keras.datasets.mnist.load_data()
         X_{\text{train}} = X_{\text{train.reshape}}(60000, 784)
         X_{\text{test}} = X_{\text{test.reshape}}(10000, 784)
         X_train = X_train[:5000, :]
         y_train = y_train[:5000]
         X_test = X_test[:100, :]
         y_{\text{test}} = y_{\text{test}}[:100]
         model = svm.SVC()
         model.fit(X_train, y_train)
         y_pred = model.predict(X_test)
         indexToCompare = 6
         title = 'True: ' + str(y_test[indexToCompare]) + ', Prediction: ' + str(y_pred[indexToCompare])
         plt.title(title)
         plt.imshow(X_test[indexToCompare].reshape(28,28), cmap='gray')
         plt.grid(None)
         plt.axis('off')
         plt.show()
```

True: 4, Prediction: 4



```
In []: acc = metrics.accuracy_score(y_test, y_pred)
    print('\nAccuracy: ', acc)

    digits = pd.DataFrame.from_dict(y_train)

    ax = sns.countplot(x=0, data=digits)

    ax.set_title("Distribution of Digit Images in Test Set")
    ax.set(xlabel='Digit')
    ax.set(ylabel='Count')

    plt.show()

cm = metrics.confusion_matrix(y_test, y_pred)
    ax = plt.subplots(figsize=(9, 6))

sns.heatmap(cm, annot=True)

plt.show()
```

Accuracy: 0.98



