In [1]:
```
from tensorflow.keras import datasets
```

In [2]:
```
(x_train,y_train),(x_test,y_test)=datasets.mnist.load_data()
```

In [3]:
```
x_train.shape
```

Out[3]:
```
(60000, 28, 28)
```

In [4]:
```
x_train=x_train/255
x_test=x_test/255
```

In [5]:
```
x_train=x_train.reshape(-1,28*28)
x_test=x_test.reshape(-1,28*28)
```

In [6]:
```
x_train.shape
```

Out[6]:
```
(60000, 784)
```

In [7]:
```
from sklearn.svm import SVC

model=SVC()
```

In [8]:
```
model.fit(x_train,y_train)
```

Out[8]:
```
▾ SVC

SVC()
```

In [10]:
```
y_pred=model.predict(x_test)
```

In [11]:
```
from sklearn.metrics import classification_report

print(classification_report(y_test,y_pred))
```

```
              precision    recall  f1-score   support

           0       0.98      0.99      0.99       980
           1       0.99      0.99      0.99      1135
           2       0.98      0.97      0.98      1032
           3       0.97      0.99      0.98      1010
           4       0.98      0.98      0.98       982
           5       0.99      0.98      0.98       892
           6       0.99      0.99      0.99       958
           7       0.98      0.97      0.97      1028
           8       0.97      0.98      0.97       974
           9       0.97      0.96      0.97      1009

    accuracy                           0.98     10000
   macro avg       0.98      0.98      0.98     10000
weighted avg       0.98      0.98      0.98     10000
```

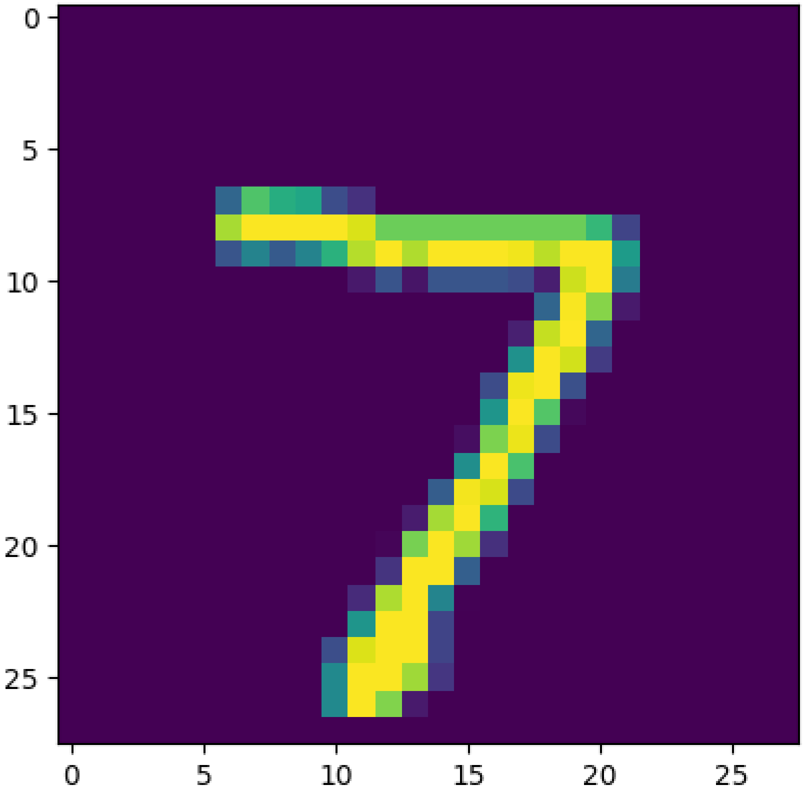
In [20]:
```
model.predict(x_test[0].reshape(-1,784))
```

```
Out[20]: array([7], dtype=uint8)
```

```
In [21]: import matplotlib.pyplot as plt

         plt.imshow(x_test[0].reshape(28,28))
```

```
Out[21]: <matplotlib.image.AxesImage at 0x25d2c7073b0>
```



```
In [ ]:
```