In [36]:

```
from keras.datasets import mnist

(train_images, train_labels), (test_images, test_labels) = mnist.load_data()
```

In [37]:

```
train_images.shape
```

Out[37]:

```
(60000, 28, 28)
```

In [38]:

```
len(train_labels)
```

Out[38]:

```
60000
```

In [39]:

```
train_labels
```

Out[39]:

```
array([5, 0, 4, ..., 5, 6, 8], dtype=uint8)
```

In [40]:

```
test_images.shape
```

Out[40]:

```
(10000, 28, 28)
```

In [41]:

```
len(test_labels)
```

Out[41]:

```
10000
```

In [42]:

```
test_labels
```

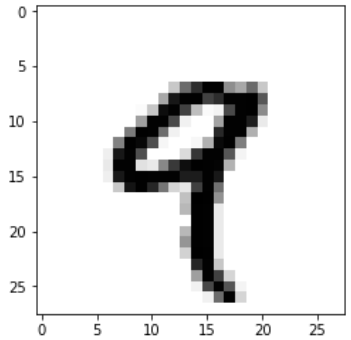
Out[42]:

```
array([7, 2, 1, ..., 4, 5, 6], dtype=uint8)
```

In [43]:

```
import matplotlib.pyplot as plt

digit = train_images[4]
plt.imshow(digit, cmap=plt.cm.binary)
plt.show()
```



In [44]:

```
from keras import models
from keras import layers

network = models.Sequential()
network.add(layers.Dense(512, activation='relu', input_shape=(28 * 28,)))
network.add(layers.Dense(10, activation='softmax'))
```

In [45]:

```
network.compile(optimizer='rmsprop',
                loss='categorical_crossentropy',
                metrics=['accuracy'])
```

In [46]:

```
train_images = train_images.reshape((60000, 28 * 28))
train_images = train_images.astype('float32') / 255

test_images = test_images.reshape((10000, 28 * 28))
test_images = test_images.astype('float32') / 255
```

In [47]:

```
from keras.utils import to_categorical

train_labels = to_categorical(train_labels)
test_labels = to_categorical(test_labels)
```

In [48]:

```
network.fit(train_images, train_labels, epochs=5, batch_size=128)
```

```
WARNING:tensorflow:From C:WUsersWJWAnaconda3WlibWsite-packagesWtensorflowWpythonWo
psWmath_ops.py:3066: to_int32 (from tensorflow.python.ops.math_ops) is deprecated
and will be removed in a future version.
Instructions for updating:
Use tf.cast instead.
Epoch 1/5
60000/60000 [==============================] - 6s 99us/step - loss: 0.2536 - acc:
0.9264
Epoch 2/5
60000/60000 [==============================] - 1s 23us/step - loss: 0.1031 - acc:
0.9700
Epoch 3/5
60000/60000 [==============================] - 1s 23us/step - loss: 0.0685 - acc:
0.9797
Epoch 4/5
60000/60000 [==============================] - 1s 23us/step - loss: 0.0502 - acc:
0.9849
Epoch 5/5
60000/60000 [==============================] - 1s 23us/step - loss: 0.0377 - acc:
0.9888
```

Out[48]:

```
<keras.callbacks.History at 0x216b7e61978>
```

In [49]:

```
test_loss, test_acc = network.evaluate(test_images, test_labels)
```

```
10000/10000 [==============================] - 0s 35us/step
```

In [50]:

```
print('test_acc:', test_acc)
```

```
test_acc: 0.9827
```