In [1]:
```
import imageio
```

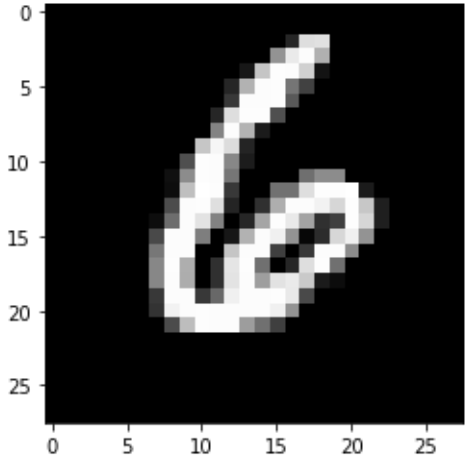
In [2]:
```
img = imageio.imread('image6_1.png')
```

In [3]:
```
import numpy as np
```

In [4]:
```
gray = np.dot(img[..., :3], [0.299, 0.587, 0.114])
```

In [5]:
```
import matplotlib.pyplot as plt
```

In [6]:
```
plt.imshow(gray, cmap = plt.get_cmap('gray'))
plt.show()
```




In [7]:
```
gray = gray.reshape(-1, 28, 28, 1)
gray = gray / 255
```

In [8]:
```
from keras.models import load_model
model = load_model('model.h5')
```

```
Using TensorFlow backend.
```

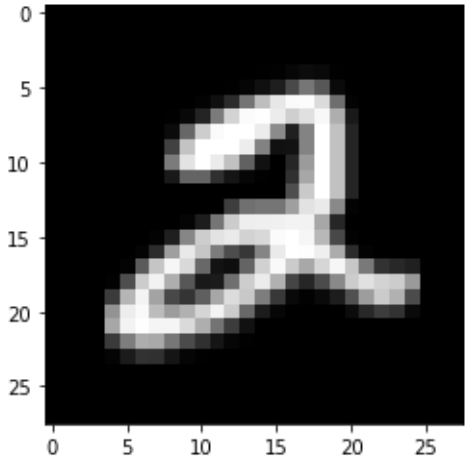
In [9]:
```
prediction = model.predict(gray)
```

In [10]:
```
print('Predicted value:', prediction.argmax())
```

```
Predicted value: 6
```

In [11]:
```
img = imageio.imread('image2.png')
gray = np.dot(img[..., :3], [0.299, 0.587, 0.114])
```

In [12]:
```
plt.imshow(gray, cmap = plt.get_cmap('gray'))
plt.show()
```



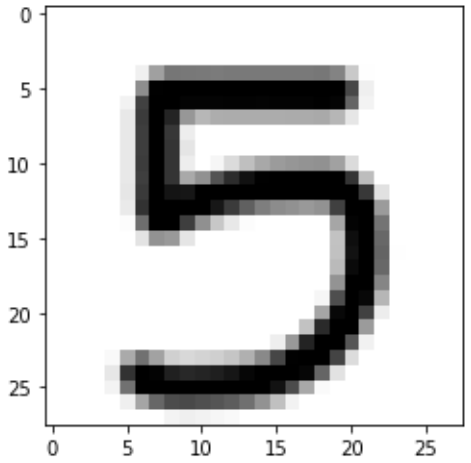

In [13]:
```
gray = gray.reshape(-1, 28, 28, 1)
gray = gray / 255
```

In [14]:
```
prediction = model.predict(gray)
```

In [15]:
```
print('Predicted value:', prediction.argmax())
```

```
Predicted value: 2
```

In [16]:
```
img = imageio.imread('image5_2.png')
gray = np.dot(img[..., :3], [0.299, 0.587, 0.114])
plt.imshow(gray, cmap = plt.get_cmap('gray'))
plt.show()
```




In [17]:
```
gray = gray.reshape(-1, 28, 28, 1)
gray = gray / 255
prediction = model.predict(gray)
print('Predicted value:', prediction.argmax())
```

```
Predicted value: 5
```

In [ ]: