In [2]:

```
import numpy as np
import pandas as pd
import seaborn as sns
import  matplotlib.pyplot as plt
from scipy.stats import geom,binom
```

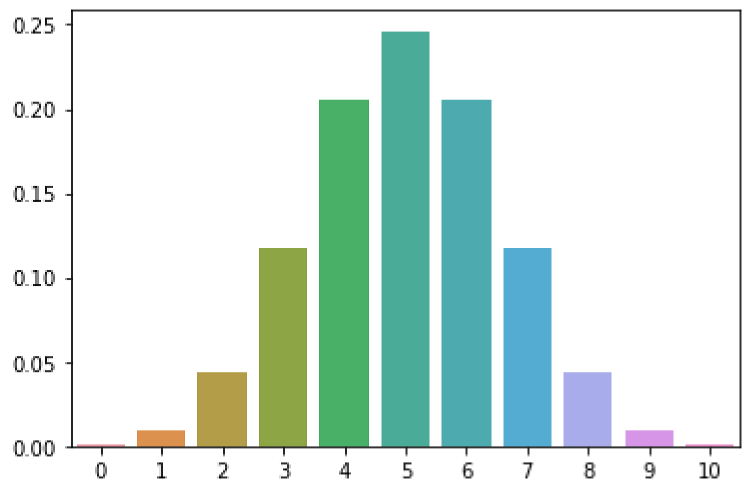
Hypothesis Testing

In [3]:

```
x_val=np.arange(0,11)
y_val=binom.pmf(n=10,k=x_val,p=0.5)
sns.barplot(x=x_val,y=y_val)
```

Out[3]:

```
<AxesSubplot:>
```



In [4]:

```
1-binom.cdf(n=10,k=6,p=0.5)
```

Out[4]:

```
0.171875
```

In [5]:

```
binom.pmf(n=10,k=10,p=0.5)+binom.pmf(n=10,k=7,p=0.5)+binom.pmf(n=10,k=8,p=0.5)+binom.pmf(n=10,k=9,p=0.5)
```

Out[5]:

```
0.17187499999999994
```

In [6]:

```
1-binom.cdf(n=100,k=69,p=0.5)
```

Out[6]:

```
3.925069822796612e-05
```

In [7]:

```
1-binom.cdf(n=1000,k=699,p=0.5)
```

Out[7]:

```
0.0
```

In [94]:

```
1-binom.cdf(n=1000,k=699,p=0.7)
```

Out[94]:

```
0.5155935198141187
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [83]:

```
pd.value_counts(np.random.choice(["H","T"],size=100000))
```

Out[83]:

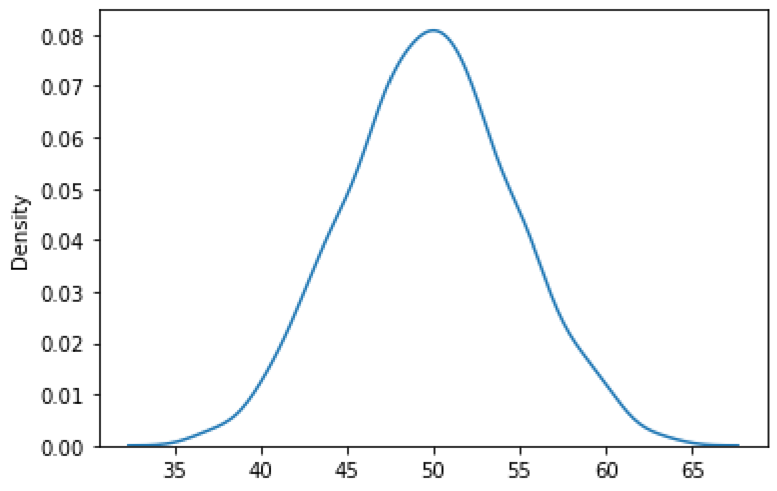
```
T    50046
H    49954
dtype: int64
```

In [93]:

```
h_val=[]
for i in range(1000):
    size=100
    h_val.append(pd.value_counts(np.random.choice(["H","T"],size=size))["H"])
sns.kdeplot(h_val)
```

Out[93]:

```
<AxesSubplot:ylabel='Density'>
```



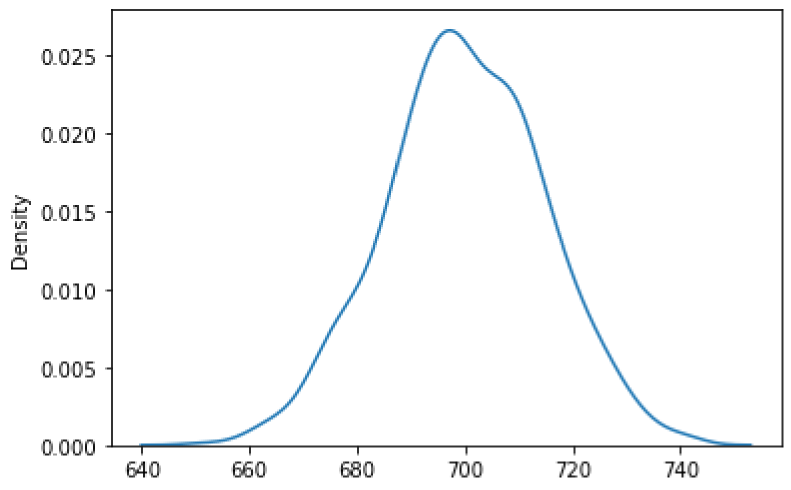
In [ ]:

In [97]:

```
h_val=[]
for i in range(1000):
    size=1000
    h_val.append(pd.value_counts(np.random.choice(["H","T"],p=[0.7,0.3],size=size))["H"])
sns.kdeplot(h_val)
```

Out[97]:

```
<AxesSubplot:ylabel='Density'>
```



In [95]:

```
np.random.choice?
```

In [ ]:

In [ ]:

In [ ]:

In [ ]: