In [5]:
```
#Probability of getting 3 when a die is rolled
    ns = {1,2,3,4,5,6}
    na = {3}
    pa = len(na)/len(ns)
    print("Probability of getting 3 is:", pa)
```

```
Probability of getting 3 is: 0.16666666666666666
```

In [6]:
```
#Calculate the probabillity of atleast getting one head when coin is tossed thrice

ns={'HHH','THH','HTH','HHT','TTH','THT','HTT','TTT'}
na={'HHH','THH','HTH','HHT','TTH','THT','HTT'}
pa=len(na)/len(ns)
print("Probability of atleast getting one head is:",pa)
```

```
Probability of atleast getting one head is: 0.875
```

In [7]:
```
#Glass of jar contain 5 red,3 blue and 2 green jelly beans. If a jelly is chosen at random from jar, what is probability that it is not a blue
ns=10
na=7
pa=na/ns
print("Probability of not getting blue jar is:",pa)
```

```
Probability of not getting blue jar is: 0.7
```

In [8]:
```
 P=0.7*0.5
print("Probability that they will be alive after 20 years is:",P)
```

```
Probability that they will be alive after 20 years is: 0.35
```

In [9]:
```
def probability(number_of_events,samplespace):
    return number_of_events/samplespace
```

In [11]:
```
pa=probability(2,6)
pb=probability(3,6)
P=pa*pb
print("Probability of getting a 4 or 5 on the first toss and 1, 2 or 3 on second toss is:", P)
```

```
Probability of getting a 4 or 5 on the first toss and 1, 2 or 3 on second toss is: 0.16666666666666666
```

In [12]:
```
pa=probability(5,10)
pb=probability(3,9)
pc=probability(2,8)
P=pa*pb*pc
print("The probability of obtaining white, black and green in the order is:",P)
```

```
The probability of obtaining white, black and green in the order is: 0.041666666666666664
```

In [13]:
```
cards=52
hearts=13
clubs=13
heart_or_club=probability(13,52)+probability(13,52)
print('Probability fo drawing heart or club in a deck of 52 cards is',heart_or_club)
```

```
Probability fo drawing heart or club in a deck of 52 cards is 0.5
```

In [14]:
```
cards=52
ace=4
king=4
queen=4
ace_or_king_or_queen=probability(4,52)+probability(4,52)+probability(4,52)
print("Probability of drawing ace, king or queen cards from deck is:",ace_or_king_or_queen)
```

```
Probability of drawing ace, king or queen cards from deck is: 0.23076923076923078
```

In [15]:
```
#Probability of getting heart or ace
heart=13
ace=4
ace_of_hearts=1
ha=probability(13,52)+probability(ace,52)-probability(ace_of_hearts,cards) #Additive rule
print(ha)
```

```
0.3076923076923077
```

In [16]:
```
#Probability of not getting 5 when a fair die is rolled
ns=6 #n(s)= {1,2,3,4,5,6}
na=1 #n(a)={5}
pa=na/ns
print('probability of not getting 5 is ', 1-pa)
```

```
probability of not getting 5 is  0.8333333333333334
```

Conditional Probability

In [17]:
```
import pandas as pd
import numpy as np
df=pd.read_csv('F:/MSC 1/FDS/notes/student-mat.csv') #Student-mat.csv file
df.head(3)
```

Out[17]:

	school	sex	age	address	famsize	Pstatus	Medu	Fedu	Mjob	Fjob	...	famrel	freetime	goout	Dalc	Walc	health	absences	G1	G2	G3
0	GP	F	18	U	GT3	A	4	4	at_home	teacher	...	4	3	4	1	1	3	6	5	6	6
1	GP	F	17	U	GT3	T	1	1	at_home	other	...	5	3	3	1	1	3	4	5	5	6
2	GP	F	15	U	LE3	T	1	1	at_home	other	...	4	3	2	2	3	3	10	7	8	10

3 rows × 33 columns

In [18]:
```
 len(df)
```

Out[18]:
```
395
```

In [19]:
```
df['grade_A']=np.where(df['G3']*5 >= 80 , 1, 0)
```

If Student is absent more then 10 or more classes then they should be in the high_absences list

In [20]:
```
df['high_absences']=np.where(df['absences']>=10,1,0)
```

Add one more column to make building a pivot table easier

In [21]:
```
df['count']=1
```

Drop all other columns and just display new added 3 columns

In [22]:
```
df=df[['grade_A','high_absences','count']]
df.head()
```

Out[22]:

	grade_A	high_absences	count
0	0	0	1
1	0	0	1
2	0	1	1
3	0	0	1
4	0	0	1

In [23]:
```
final=pd.pivot_table(df,values='count',index=['grade_A'],columns=['high_absences'],
aggfunc=np.size,fill_value=0)
```

In [24]:
```
 print(final)
```

```
high_absences    0   1
grade_A
0              277  78
1               35   5
```

In [25]:
```
#probability of a grade of 80% or greater

Pa = (35 + 5) / (35 + 5 + 277 + 78)
print(Pa)
```

```
0.10126582278481013
```

In [26]:
```
Pb = (78 + 5) / (35 + 5 + 277 + 78)
print(Pb)
```

```
0.21012658227848102
```

In [27]:
```
PaAndb = 5 / (35 + 5 + 277 + 78)
print(PaAndb)
```

```
0.012658227848101266
```

In [28]:
```
 print(PaAndb / Pb)
```

```
0.060240963855421686
```

In [ ]:
```

```