

*Program:-

```
import collections  
import matplotlib.pyplot as plt  
  
def Plot_dice_Probabilities():  
    total_outcomes = 6*6  
    sums = []  
  
    for die1 in range(1,7):  
        for die2 in range(1,7):  
            sums.append(die1+die2)  
  
    frequency = collections.Counter(sums)  
  
    sums_sorted = sorted(frequency.keys())  
    frequencies_sorted = [frequency[s] for s in sums_sorted]  
  
    Probabilities = [freq/total_outcomes for freq in frequencies_sorted]  
  
    print(" --- Dice Roll Probability ---")  
    print(f"\nTotal Possible outcomes : {total_outcomes}")  
    print("\nSum | Frequency | Probability")  
  
    ale = set_xlabel("Sum of Two Dice", fontsize=14)  
    ale = set_ylabel("Probability", fontsize=14)
```

Date: 20/10/25 USE Case -2

Aim: To develop a python application that calculates and visualizes the probabilities of different outcomes when rolling a pair of standard six-sided dice.

Algorithm:-

1. Start
2. Define the sample space : Determine the total number of possible outcomes when rolling two dice. Since each die has 6 faces, the total is $6 \times 6 = 36$.
3. Calculate frequencies : Iterate through all possible combinations of two dice.
4. Calculate probabilities : Divide the frequency of each sum by the total number of outcomes.
5. Visualize : use the Matplotlib library to create a bar chart:
The x-axis will represent sum.
The y-axis will represent probabilities.
6. Stop.

ax.set_xticks(sums_sorted)

ax.set ylim(0, max(probabilities) * 1.1)

for bar in bars:

height = bar.get_height()

ax.text(bar.get_x() + bar.get_width(),

height + 0.005,

f'height: .4f}',

ha='center',

baseline='bottom',

fontsize=10)

Plot-dice-probabilities()

Outputs

Sum	Frequency	Probabilities
2	1	0.0278
3	2	0.0556
4	3	0.0883
5	4	0.1111
6	5	0.1389
7	6	0.1667
8	5	0.1389
9	4	0.1111
10	3	0.0883
11	2	0.0556
12	1	0.0278

VEL TECH	
EX No.	05
PERFORMANCE (S)	05
RESULT AND ANALYSIS (S)	05
DISCUSSION (S)	04
REPORT (S)	04
FINAL (S)	04
SIGN WITH DATE	14/12/2023

Result: Hence, the execution of Python code is generated successfully.