

### \* Program:-

```
import collections
import matplotlib.pyplot as plt

def plot_dice_probabilities():
    total_outcome = 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_outcome for freq in frequencies_sorted]

    print("\n --- Dice Roll Probability ---")
    print(f"Total possible outcomes: {total_outcome}")
    print("\n Sum | Frequency | Probability")
    ax = plt.gca()
    ax.set_xlabel("Sum of Two Dice", fontsize=14)
    ax.set_ylabel("Probability", fontsize=14)
```

Date:- 20/10/25 USE Case - 2

Aim:- To develop a python application that calculates and visualizes the probability 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 the sum. The y-axis will represent the 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',
```

```
            va='bottom',
```

```
            fontsize=6)
```

```
plt.dice_probabilities()
```

Output:

Sum	Frequency	Probabilities
2	1	0.0278
3	2	0.0556
4	3	0.0833
5	4	0.1111
6	5	0.1389
7	6	0.1667
8	5	0.1389
9	4	0.1111
10	3	0.0833
11	2	0.0556
12	1	0.0278

VELTECH	
Sl. No.	
PERFORMANCE (%)	05
RESULT AND ANALYSIS (%)	05
VIVA VOCE (%)	05
RECORD (%)	05
TOTAL (%)	
DATE WITH DATE	

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