EXPERIMENT NO: 11

Random Sampling and Sampling Distribution

Aim:

To explore random sampling from a population and understand the concept of sampling distribution using Python in Jupyter Notebook.

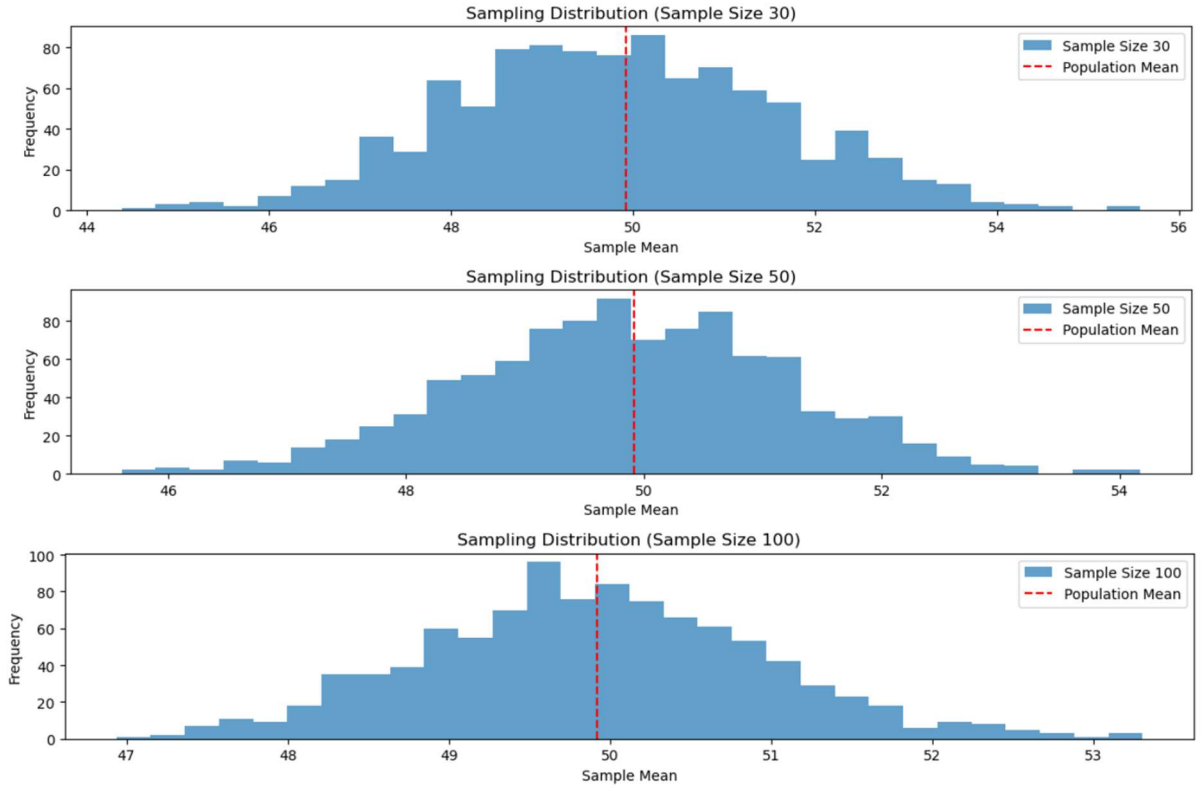
Algorithm:

- **Initialize population parameters**
- **Generate the population**
- **Select sample sizes**
- **Repeat sampling**
- **Compute sample means**
- **Form sampling distributions**
- **Plot histograms**
- **Mark population mean**
- **Analyze results**

Code:

```python
import numpy as np
import matplotlib.pyplot as plt


population_mean = 50
population_std = 10
population_size = 100000
population = np.random.normal(population_mean, population_std, population_size)


sample_sizes = [30, 50, 100]
num_samples = 1000
sample_means = {}

for size in sample_sizes:
    sample_means[size] = []

    for _ in range(num_samples):
        sample = np.random.choice(population, size=size, replace=False)
        sample_means[size].append(np.mean(sample))


plt.figure(figsize=(12, 8))

for i, size in enumerate(sample_sizes):
    plt.subplot(len(sample_sizes), 1, i+1)
    plt.hist(sample_means[size], bins=30, alpha=0.7, label=f'Sample Size {size}')
    plt.axvline(np.mean(population), color='red', linestyle='dashed', linewidth=1.5, label='Population Mean')
    plt.title(f'Sampling Distribution (Sample Size {size})')
    plt.xlabel('Sample Mean')
    plt.ylabel('Frequency')
    plt.legend()
```

```python
plt.tight_layout()
plt.show()
```



Result:

All sampling distributions are centered around 50, and as the sample size increases, the spread decreases, giving more accurate estimates of the population mean.