

Code 1

```
[3]: def sample_normal_dist(mu, sigma, n, plot=True):
      # Generate random samples from a normal distribution
      samples = np.random.normal(loc=mu, scale=sigma, size=n)
      if plot:
          # Visualize the distribution of the samples using a histogram
          plt.hist(samples, bins=30, density=True, alpha=0.6, color='g')
          plt.title('Histogram of Samples from Normal Distribution')
          plt.xlabel('Value')
          plt.ylabel('Probability Density')
          plt.grid(True)
          plt.show()

      # Display the mu and sigma
      df = pd.DataFrame({"True": [mu, sigma],
                        "Sample": [np.mean(samples), np.std(samples)]},
                        index=["Mu", "Sigma"])
      display(df.round(3))
      return
```

```
* [8]: mu = 0
      sigma = 1
      n = 1000
      sample_normal_dist(mu, sigma, n)
```

```
* [9]: mu = 1
      sigma = 4
      n = 500
      sample_normal_dist(mu, sigma, n)
```

Code 2

```
[2]: # Plot normal with mu=0, sigma=1, n=1000
      mu = 0
      sigma = 1
      n = 1000
      # Generate random samples from a normal distribution
      samples = np.random.normal(loc=mu, scale=sigma, size=n)

      # Visualize the distribution of the samples using a histogram
      plt.hist(samples, bins=30, density=True, alpha=0.6, color='g')
      plt.title('Histogram of Samples from Normal Distribution')
      plt.xlabel('Value')
      plt.ylabel('Probability Density')
      plt.grid(True)
      plt.show()

      # Display the mu and sigma
      df = pd.DataFrame({"True": [mu, sigma],
                        "Sample": [np.mean(samples), np.std(samples)]},
                        index=["Mu", "Sigma"])
      display(df.round(3))
      # Plot normal with mu=1, sigma=4, n=500
      mu = 1
      sigma = 4
      n = 500
      # Generate random samples from a normal distribution
      samples = np.random.normal(loc=mu, scale=sigma, size=n)

      # Visualize the distribution of the samples using a histogram
      plt.hist(samples, bins=30, density=True, alpha=0.6, color='g')
      plt.title('Histogram of Samples from Normal Distribution')
      plt.xlabel('Value')
      plt.ylabel('Probability Density')
      plt.grid(True)
      plt.show()

      # Display the mu and sigma
      df = pd.DataFrame({"True": [mu, sigma],
                        "Sample": [np.mean(samples), np.std(samples)]},
                        index=["Mu", "Sigma"])
      display(df.round(3))
```

Code 1

```
def sample_normal_dist(mu,sigma, n, plot =True):
    ~~~~~
    # Generate random samples from a normal distribution
    samples =np.random.normal( loc= mu, scale = sigma, size=n)
    if plot:
        # Visualize the distribution of the samples using a histogram
        plt.hist(samples,bins= 30, density =True,alpha= 0.6,color='g' )
        plt.title('Histogram of Samples from Normal Distribution' )
        plt.xlabel( 'Value')

    ~~~~~
    plt.ylabel('Probability Density')
    plt.grid(True)
    plt.show()

    ~~~~~
    # Display the mu and sigma
    df= pd.DataFrame({"True": [mu,
                                sigma], "Sample": [np.mean(samples),
                                                    np.std(samples)]},
                    index=["Mu", "Sigma"])
    display(df.round(3))
    return samples
```

Code 2

```
def sample_normal_dist(mu: float, sigma: float, n: int, plot=True):
    """
    Samples from a normal distribution with given parameters with
    the option to plot

    Args:
        mu (float): mean.
        sigma (float): standard deviation.
        n (int): number of samples.

    Returns:
        np.array: array of samples.
    """
    # Generate random samples from a normal distribution
    samples = np.random.normal(loc=mu, scale=sigma, size=n)
    if plot:
        # Visualize the distribution of the samples using a histogram
        plt.hist(samples, bins=30, density=True, alpha=0.6, color='g')
        plt.title('Histogram of Samples from Normal Distribution')
        plt.xlabel('Value')
        plt.ylabel('Probability Density')
        plt.grid(True)
        plt.show()

    # Display the mu and sigma
    df = pd.DataFrame({"True": [mu, sigma],
                            "Sample": [np.mean(samples), np.std(samples)]},
                    index=["Mu", "Sigma"])
    display(df.round(3))
    return samples
```