

LEGENDRE FUNCTION

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

def legendre(n, x):

    if n == 0:
        Pnx = 1
    elif n == 1:
        Pnx = x
    else:
        p = (n / (n+1)) * legendre(n-2, x)

        Pnx = ((2*n+1) / (n+1)) * x * legendre(n-1, x) - p

    return Pnx

N = [0, 1, 2, 3, 4, 5]

X = np.linspace(-1, 1, 1000)

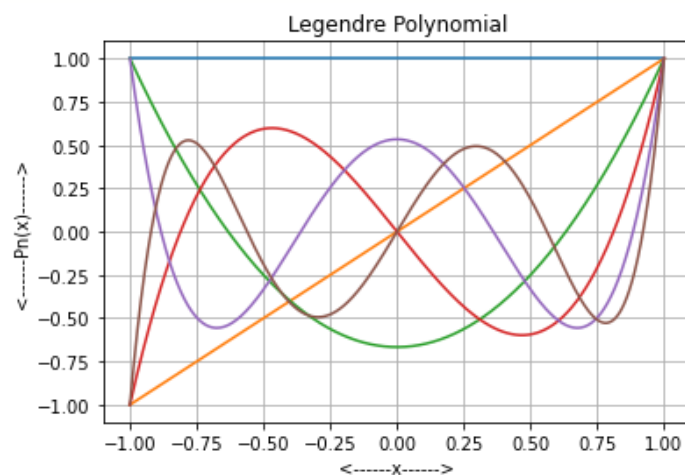
for a in range(len(N)):

    P = []
    for i in range(len(X)):
        P.append(legendre(a, X[i]))

    plt.plot(X, P)

plt.grid(True)
plt.xlabel('<-----x----->')
plt.ylabel('<-----Pn(x)----->')
plt.title('Legendre Polynomial')
plt.show()
```

OUTPUT -



```

# Using pre-defined Legendre function

from scipy.special import legendre

x = np.linspace(-1, 1, 1000)

def Legendre(n, x):
    leg = legendre(n)
    P_n = leg(x)

    return P_n

N = [0, 1, 2, 3, 4, 5]

for a in range(len(N)):
    P = Legendre(a, x)

    plt.plot(x, P)

plt.title('In-Built LEGENDRE FUNCTION', fontweight = 'bold')
plt.xlabel('x', fontweight = 'bold', fontsize = 16)
plt.ylabel('Pn(x)', fontweight = 'bold', fontsize = 16)
plt.grid()
plt.show()

```

OUPTUT –

