Experiment 1a:

Aim: \ To generate Legendre Polynomials using Rodrigues' formula and plot the first six Legendre polynomials .

Software packages used: \ language- Python3, packages- Numpy, Matplotlib, Sympy

Theory: \ The Rodrigues formula for Legendre Polynomials is given by: \

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n}(x^2 - 1)^n$$

Code:

- Importing packages

In [1]:
```
import numpy as np
import matplotlib.pyplot as plt
from sympy import *
```

- Defining a function for Rodrigues formula

In [2]:
```
def legendre (n,y):
    x = symbols('x')
    dif = diff((x**2 -1)**n,x,n)
    output = (1/(2**n * np.math.factorial(n))) * dif.subs(x,y)
    return float(output)

# this allows the defined function to take array input for y argument
legendre_polynomial = np.vectorize(legendre)
```

- generating 1000 points in the interval [-1,1] on the X-axis

In [3]:
```
X = np.linspace(-1,1,1000)
```

- generating legendre polynomial y values for corresponding x values

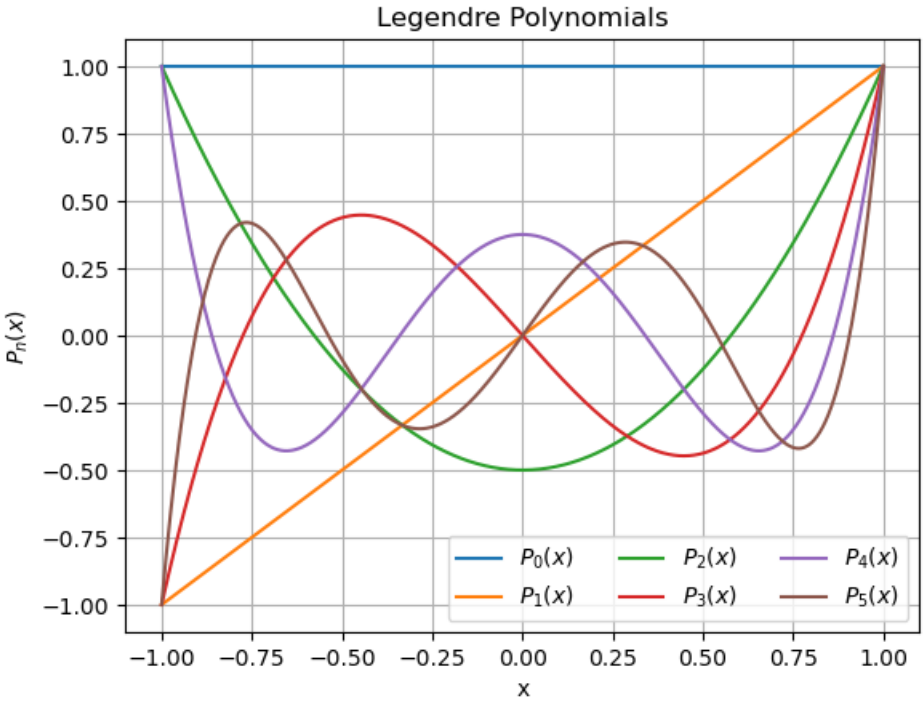
In [4]:
```
P0 = legendre_polynomial(0,X)
P1 = legendre_polynomial(1,X)
P2 = legendre_polynomial(2,X)
P3 = legendre_polynomial(3,X)
P4 = legendre_polynomial(4,X)
P5 = legendre_polynomial(5,X)
```

- plotting generated data

In [5]:
```
plt.plot(X,P0,label='$P_0(x)$')
plt.plot(X,P1,label='$P_1(x)$')
plt.plot(X,P2,label='$P_2(x)$')
plt.plot(X,P3,label='$P_3(x)$')
plt.plot(X,P4,label='$P_4(x)$')
plt.plot(X,P5,label='$P_5(x)$')

plt.title("Legendre Polynomials")
plt.xlabel("x")
plt.ylabel("$P_n(x)$")
plt.grid()
plt.legend(loc='best',ncol = 3)

plt.show()
```


Legendre Polynomials
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
$P_n(x)$
−1.00
−0.75
−0.50
−0.25
0.00
0.25
0.50
0.75
1.00
x
$P_0(x)$
$P_1(x)$
$P_2(x)$
$P_3(x)$
$P_4(x)$
$P_5(x)$