Create a class to store the information of a single input mathematical function, e.g. $f(x) = x^3 + 3x^2 - x + 3$

In [67]:
```
class Function:
    # Coefficients are considered to be zero for empty terms, e.g.
    # f(x) = x^2 + 3 would have coefficients = [3, 0, 1]
    def __init__(self, coefficients):
        self.coefficients = coefficients


    def evaluate(self, x):
        if self.coefficients is None:
            return 0
        total = 0
        for i in range(len(self.coefficients)):
            total += self.coefficients[i] * x ** i
        return total


    def differentiate(self, n):
        if self.coefficients is None or n > len(self.coefficients):
            if n > len(self.coefficients):
                self.coefficients = None
            return
        for i in range(len(self.coefficients)):
            self.coefficients[i] *= i
        for i in range(n):
            del self.coefficients[0]
```

Now create the function $f(x) = x$

In [73]:
```
f = Function([0, 1])
```

Show that $f(0) = 0$ and that $f(1) = 1$

In [75]:
```
print(f.evaluate(0))
print(f.evaluate(1))
```

```
0
1
```

Now take the 3rd derivative of f with respect to x and then show that $f'''(n) = 0$

In [76]:
```
f.differentiate(3)
print(f.evaluate(50))
print(f.evaluate(1000))
```

```
0
0
```

In [ ]:
```

```