

```
In [3]: from __future__ import print_function
%matplotlib inline
#import ganymede
#ganymede.configure('uav.beaver.works')
import matplotlib.pyplot as plt
import numpy as np
import sympy as sym
sym.init_printing(use_latex = "mathjax")
```

```
-----
ModuleNotFoundError                                Traceback (most recent call last)
/tmp/ipykernel_8253/2461489746.py in <module>
      1 from __future__ import print_function
      2 get_ipython().run_line_magic('matplotlib', 'inline')
----> 3 import ganymede
      4 ganymede.configure('uav.beaver.works')
      5 import matplotlib.pyplot as plt

ModuleNotFoundError: No module named 'ganymede'
```

Enter your name below and run the cell:

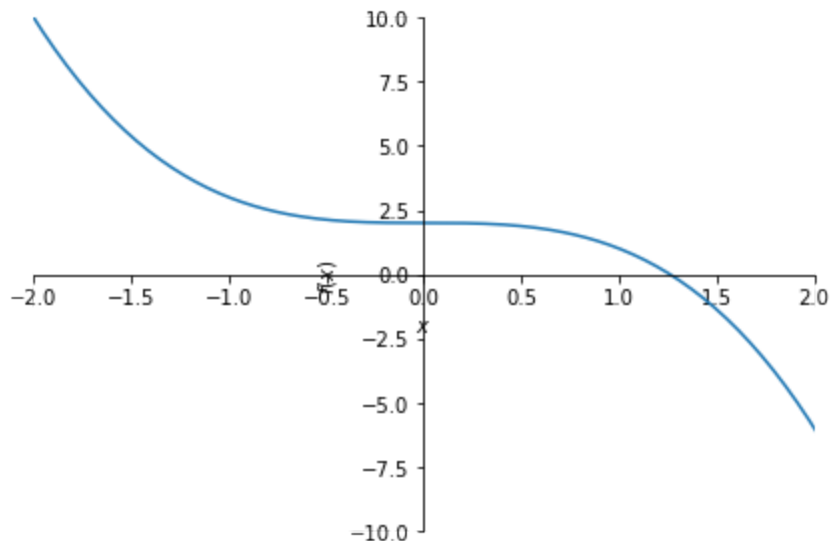
Individual cells can be run with **Ctrl** + **Enter**

In []:

```
In [4]: x = sym.symbols('x')
expr = -x ** 3 + 2

sym.plot(expr, xlim=(-2, 2), ylim=(-10, 10));
```

```
/usr/lib/python3/dist-packages/scipy/__init__.py:146: UserWarning: A NumPy version >
=1.17.3 and <1.25.0 is required for this version of SciPy (detected version 1.26.4
warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}")
```



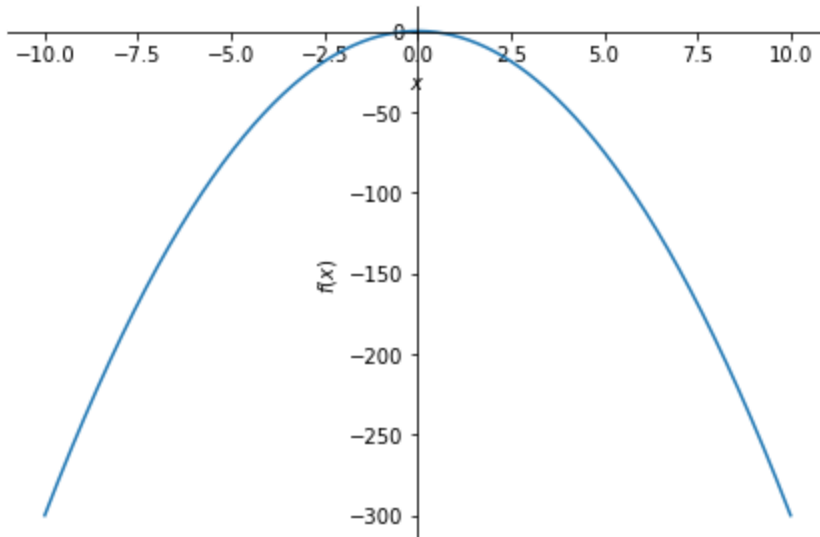
```
In [5]: sym.Derivative(expr)
```

Out[5]: $\frac{d}{dx}(2 - x^3)$

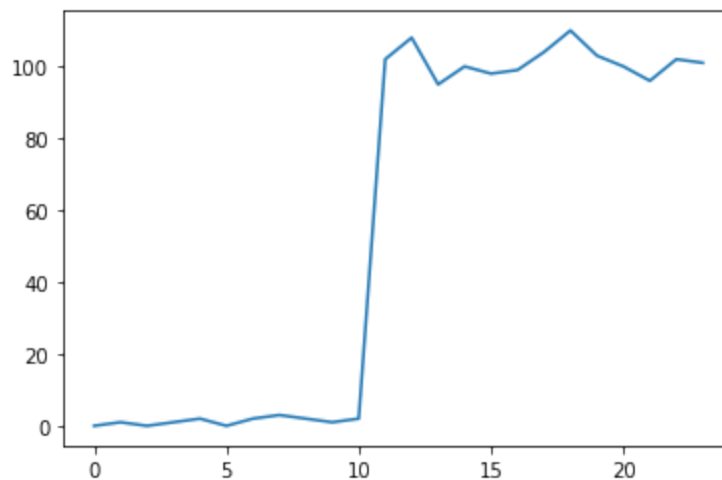
```
In [6]: sym.Derivative(expr).doit()
```

Out[6]: $-3x^2$

```
In [7]: sym.plot(sym.diff(expr));
```



```
In [8]: ys = np.array([0, 1, 0, 1, 2, 0, 2, 3, 2, 1, 2, 102, 108, 95, 100, 98, 99, 104,
fig,ax = plt.subplots()
ax.plot([i for i in range(len(ys))], ys);
```



```
In [9]: def make_windows(sequence, windowsize):
positions = len(sequence) - windowsize + 1
windows = []
for i in range(positions):
    windows.append(sequence[i:i+windowsize])
return windows

def print_padded_seq(seq):
```

```

print("[", ",".join("{:4d}".format(i) for i in seq), ']')

def print_sliding_windows(seq, windowsize=3):
    windows = make_windows(seq, windowsize)
    for window in windows:
        print(",".join("{:4d}".format(i) for i in window))

```

```

In [11]: series = [0, 1, 0, 2, 1, 0, 1, 101, 100, 98, 102, 101]
        windowsize = 2

        print_padded_seq(series)

        print_sliding_windows(series, windowsize=windowsize)
        #check(1)

```

```

[  0,  1,  0,  2,  1,  0,  1, 101, 100,  98, 102, 101 ]
  0,  1
  1,  0
  0,  2
  2,  1
  1,  0
  0,  1
  1, 101
101, 100
100,  98
 98, 102
102, 101

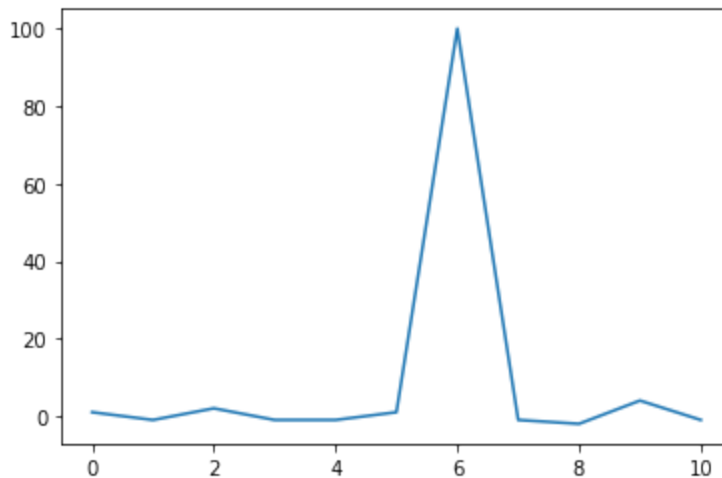
```

```

In [13]: convolutions = []
        kernel = np.array([-1,1])
        for w in make_windows(series, windowsize=2):
            w = np.array(w)
            convolved = np.dot(w, kernel)
            convolutions.append(convolved)

        plt.plot(convolutions);
        # check(2)

```

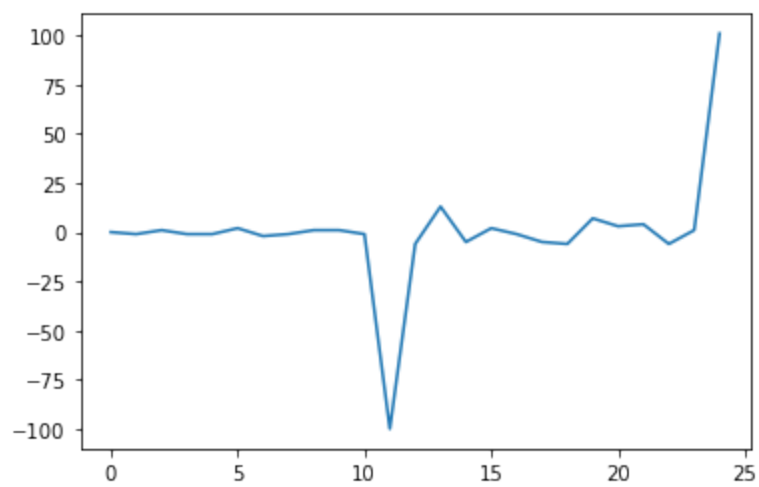


```

In [14]: convolved = np.convolve([-1, 1], ys)

```

```
fig, ax = plt.subplots()
ax.plot([i for i, _ in enumerate(convolved)], convolved);
#check(3)
```



Question: Why does the graph move up at the end?