

202001555 지은미

```
In [17]: from IPython.core.interactiveshell import InteractiveShell
InteractiveShell.ast_node_interactivity="all"
```

```
In [18]: L=list(range(10))
L2=[str(c) for c in L]
L2

type(L2[0])

L3=[True, "2", 3.0, 4]
[type(item) for item in L3]
```

Out[18]: ['0', '1', '2', '3', '4', '5', '6', '7', '8', '9']

Out[18]: str

Out[18]: [bool, str, float, int]

```
In [19]: import array
L=list(range(10))
A=array.array('i',L)
A
```

Out[19]: array('i', [0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

```
In [20]: import numpy as np
np.array([1,4,2,5,3])
np.array([3.14,4,2,3])
np.array([1,2,3,4],dtype='float32')
np.array([range(i,i+3) for i in [2,4,6]])
```

Out[20]: array([1, 4, 2, 5, 3])

Out[20]: array([3.14, 4., 2., 3.])

Out[20]: array([1., 2., 3., 4.], dtype=float32)

Out[20]: array([[2, 3, 4],
 [4, 5, 6],
 [6, 7, 8]])

```
In [21]: np.zeros(10,dtype=int)
np.ones((3,5),dtype=float)
np.full((3,5),3.14)
np.arange(0,20,2)
np.linspace(0,1,5)
np.random.random((3,3))
np.random.normal(0,1,(3,3))
```

```
np.random.randint(0,10,(3,3))

np.eye(3)

np.empty(3)
```

```
Out[21]: array([0, 0, 0, 0, 0, 0, 0, 0, 0, 0])

Out[21]: array([[1., 1., 1., 1., 1.],
 [1., 1., 1., 1., 1.],
 [1., 1., 1., 1., 1.]])]

Out[21]: array([[3.14, 3.14, 3.14, 3.14, 3.14],
 [3.14, 3.14, 3.14, 3.14, 3.14],
 [3.14, 3.14, 3.14, 3.14, 3.14]])]

Out[21]: array([ 0,  2,  4,  6,  8, 10, 12, 14, 16, 18])

Out[21]: array([0. , 0.25, 0.5 , 0.75, 1. ])

Out[21]: array([[0.65279032, 0.63505887, 0.99529957],
 [0.58185033, 0.41436859, 0.4746975 ],
 [0.6235101 , 0.33800761, 0.67475232]])]

Out[21]: array([[ 1.0657892 , -0.69993739,  0.14407911],
 [ 0.3985421 ,  0.02686925,  1.05583713],
 [-0.07318342, -0.66572066, -0.04411241]])]

Out[21]: array([[7, 2, 9],
 [2, 3, 3],
 [2, 3, 4]])

Out[21]: array([[1., 0., 0.],
 [0., 1., 0.],
 [0., 0., 1.]])

Out[21]: array([1., 1., 1.])
```

```
In [22]: import numpy as np
np.random.seed(0)

x1=np.random.randint(10, size=6)
x2=np.random.randint(10, size=(3,4))
x3=np.random.randint(10,size=(3,4,5))

print("x3 ndim: ", x3.ndim)
print("x3 shape: ",x3.shape)
print("x3 size: ", x3.size)

print("dtype: ",x3.dtype)

print("itemsize: ", x3.itemsize,"bytes")
print(" nbytes: ",x3.nbytes,"bytes")
```

```
x3 ndim: 3
x3 shape: (3, 4, 5)
x3 size: 60
dtype: int32
itemsize: 4 bytes
nbytes: 240 bytes
```

```
In [23]: x1

x1[0]

x1[4]

x1[-1]

x1[-2]
```

```
x2
x2[0,0]
x2[2,0]
x2[2,-1]
x2[0,0]=12
x2
x1[0]=3.14159
x1
```

Out[23]: array([5, 0, 3, 3, 7, 9])

Out[23]: 5

Out[23]: 7

Out[23]: 9

Out[23]: 7

Out[23]: array([[3, 5, 2, 4],
 [7, 6, 8, 8],
 [1, 6, 7, 7]])

Out[23]: 3

Out[23]: 1

Out[23]: 7

Out[23]: array([[12, 5, 2, 4],
 [7, 6, 8, 8],
 [1, 6, 7, 7]])

Out[23]: array([3, 0, 3, 3, 7, 9])

```
In [24]: x=np.arange(10)
x
x[:5]
x[5:]
x[4:7]
x[::-2]
x[1::2]
x[::-1]
x[5::-2]
```

Out[24]: array([0, 1, 2, 3, 4, 5, 6, 7, 8, 9])

Out[24]: array([0, 1, 2, 3, 4])

Out[24]: array([5, 6, 7, 8, 9])

Out[24]: array([4, 5, 6])

Out[24]: array([0, 2, 4, 6, 8])

```
Out[24]: array([1, 3, 5, 7, 9])
Out[24]: array([9, 8, 7, 6, 5, 4, 3, 2, 1, 0])
Out[24]: array([5, 3, 1])
```

```
In [25]: print(x2)

x2[:2,:3]

x2[:3,:,:2]

x2[::-1,::-1]

print(x2[:,0])

print(x2[0,:])

print(x2)

x2_sub = x2[:2,:2]
print(x2_sub)

x2_sub[0,0] = 99
print(x2_sub)

print(x2)
```

```
[[12  5  2  4]
 [ 7  6  8  8]
 [ 1  6  7  7]]
Out[25]: array([[12,  5,  2],
                 [ 7,  6,  8]])
Out[25]: array([[12,  2],
                 [ 7,  8],
                 [ 1,  7]])
Out[25]: array([[ 7,  7,  6,  1],
                 [ 8,  8,  6,  7],
                 [ 4,  2,  5, 12]])
[[12  7  1]
 [12  5  2  4]
 [12  5  2  4]
 [[12  5  2  4]
 [ 7  6  8  8]
 [ 1  6  7  7]]
 [[12  5]
 [ 7  6]]
 [[99  5]
 [ 7  6]]
 [[99  5  2  4]
 [ 7  6  8  8]
 [ 1  6  7  7]]]
```

```
In [26]: grid=np.arange(1,10).reshape((3,3))
print(grid)

x=np.array([1,2,3])

x.reshape((1,3))

x[np.newaxis,:]

x.reshape((3,1))
```

```
x[:, np.newaxis]
```

```
[[1 2 3]
 [4 5 6]
 [7 8 9]]
```

```
Out[26]: array([[1, 2, 3]])
```

```
Out[26]: array([[1, 2, 3]])
```

```
Out[26]: array([[1],
 [2],
 [3]])
```

```
Out[26]: array([[1],
 [2],
 [3]])
```

```
In [27]: x=np.array([1,2,3])
```

```
y=np.array([3,2,1])
```

```
np.concatenate([x,y])
```

```
z=[99,99,99]
```

```
print(np.concatenate([x,y,z]))
```

```
grid=np.array([[1,2,3],[4,5,6]])
```

```
np.concatenate([grid,grid],axis=1)
```

```
x=np.array([1,2,3])
```

```
grid=np.array([[9,8,7],[6,5,4]])
```

```
np.vstack([x,grid])
```

```
y=np.array([[99],[99]])
```

```
np.hstack([grid,y])
```

```
Out[27]: array([1, 2, 3, 3, 2, 1])
```

```
[ 1  2  3  3  2  1 99 99 99]
```

```
Out[27]: array([[1, 2, 3, 1, 2, 3],
 [4, 5, 6, 4, 5, 6]])
```

```
Out[27]: array([[1, 2, 3],
 [9, 8, 7],
 [6, 5, 4]])
```

```
Out[27]: array([[ 9,  8,  7, 99],
 [ 6,  5,  4, 99]])
```

```
In [28]: x=[1,2,3,99,99,3,2,1]
```

```
x1, x2, x3=np.split(x,[3,5])
print(x1,x2,x3)
```

```
grid=np.arange(16).reshape((4,4))
grid
```

```
upper, lower=np.vsplit(grid,[2])
print(upper)
print(lower)
```

```
left, right=np.hsplit(grid,[2])
print(left)
print(right)
```

```
[1 2 3] [99 99] [3 2 1]
```

```
Out[28]: array([[ 0,  1,  2,  3],
 [ 4,  5,  6,  7],
```

```
[ 8,  9, 10, 11],
[12, 13, 14, 15]])
[[0 1 2 3]
 [4 5 6 7]]
[[ 8  9 10 11]
 [12 13 14 15]]
[[ 0   1]
 [ 4   5]
 [ 8   9]
 [12 13]]
[[ 2   3]
 [ 6   7]
 [10 11]
 [14 15]]
```

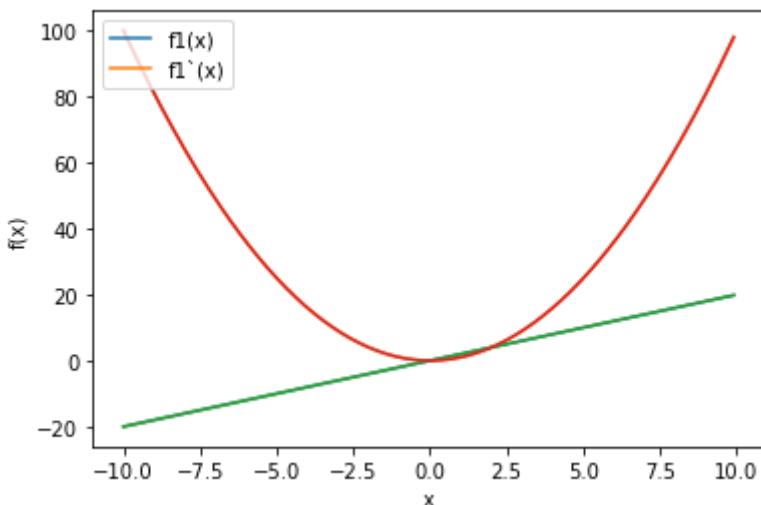
```
In [1]: import numpy as np
import matplotlib.pyplot as plt
import math
```

```
In [2]: def diff(f,x):
    h=1e-10
    return (f(x+h)-f(x))/h

def f1(x):
    return x**2

x = np.arange(-10.0,10.0,0.1)
y1 = diff(f1,x)
y2 = f1(x)
plt.xlabel("x")
plt.ylabel("f(x)")
plt.plot(x,y1,label='f1(x)')
plt.plot(x,y2,label='f1'(x))
plt.legend(loc=2)
plt.plot(x,y1)
plt.plot(x,y2)
plt.show()

print("f1'(2)=",diff(f1,2))
```



$f1'(2)= 4.000000330961484$

```
In [3]: import numpy as np
import matplotlib.pyplot as plt
import math

def diff(f,x):
    h=1e-10
    return (f(x+h)-f(x))/h
```

```

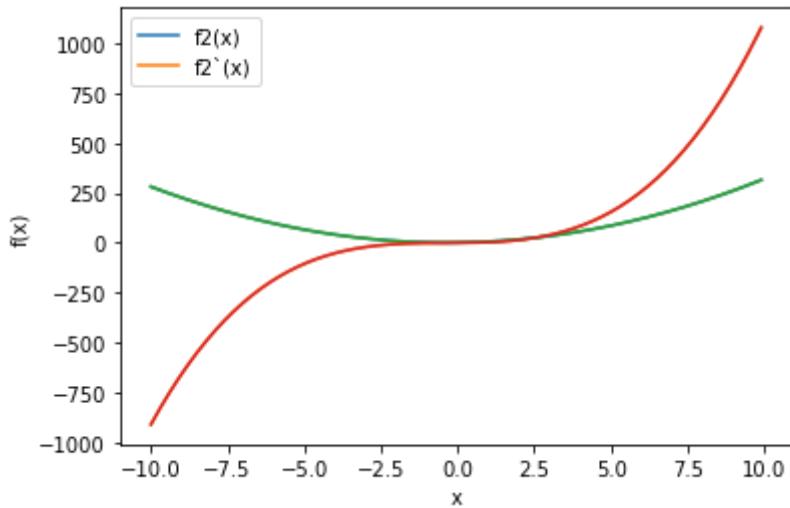
def f2(x):
    return x**3+x**2+x

x = np.arange(-10.0, 10.0, 0.1)
y1 = diff(f2,x)
y2 = f2(x)
plt.xlabel("x")
plt.ylabel("f(x)")
plt.plot(x,y1,label='f2(x)')
plt.plot(x,y2,label='f2'(x)')
plt.legend(loc=2)

plt.plot(x,y1)
plt.plot(x,y2)
plt.show()

print("f2'(2)=",diff(f2,2))

```



$f2'(2)= 17.000001406586307$

```

In [4]: import numpy as np
import matplotlib.pyplot as plt
import math

def diff(f,x):
    h=1e-10
    return (f(x+h)-f(x))/h

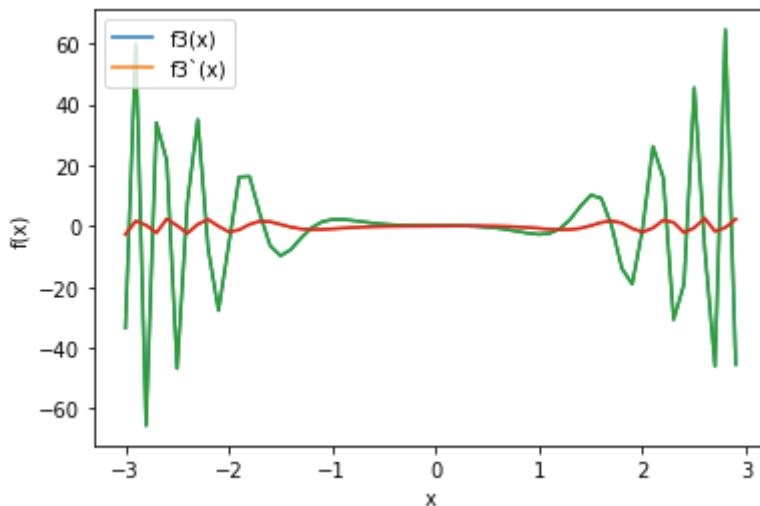
def f3(x):
    return x*np.sin(x*x*x+3)

x = np.arange(-3.0,3.0,0.1)
y1= diff(f3,x)
y2= f3(x)
plt.xlabel("x")
plt.ylabel("f(x)")
plt.plot(x,y1,label='f3(x)')
plt.plot(x,y2,label='f3'(x)')
plt.legend(loc=2)

plt.plot(x,y1)
plt.plot(x,y2)
plt.show()

print("f3'(2)=",diff(f3,2))

```



$$f3'(2) = -0.893773943744236$$

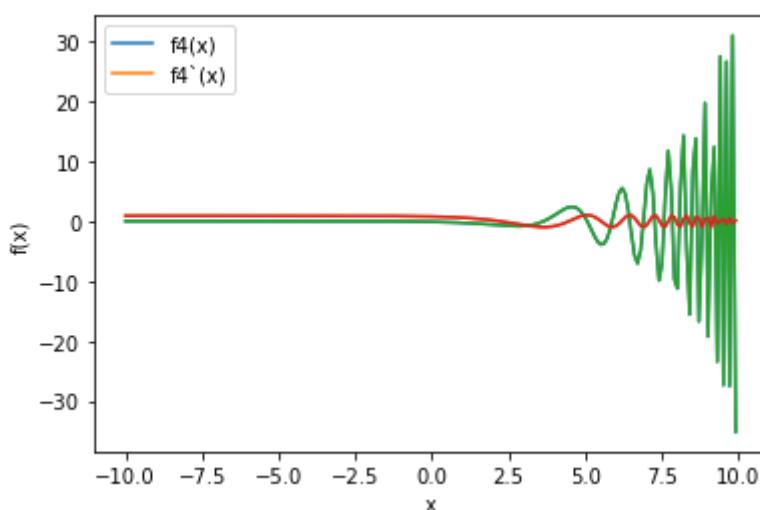
```
In [5]: import numpy as np
import matplotlib.pyplot as plt
import math
```

```
def diff(f,x):
    h=1e-10
    return (f(x+h)-f(x))/h

def f4(x):
    return np.cos(np.sqrt(np.exp(x)+1)/2)

x = np.arange(-10.0,10.0,0.1)
y1= diff(f4,x)
y2=f4(x)
plt.xlabel("x")
plt.ylabel("f(x)")
plt.plot(x,y1,label='f4(x)')
plt.plot(x,y2,label='f4'(x)')
plt.legend(loc=2)
plt.plot(x,y1)
plt.plot(x,y2)
plt.show()

print("f4'(2)=",diff(f4,2))
```



$$f4'(2) = -0.6329958779360823$$

202001555 지은미