

Example cmp pip using

!pip list

!pip freeze

conda list

pip show scipy

In [6]: !pip show opencv-python

```
Name: opencv-python
Version: 4.12.0.88
Summary: Wrapper package for OpenCV python bindings.
Home-page: https://github.com/opencv/opencv-python
Author:
Author-email:
License: Apache 2.0
Location: C:\Users\user\AppData\Local\Programs\Python\Python313\Lib\site-packages
Requires: numpy
Required-by:
```

EXAMPLE

operators & functions

In [7]: `a=3.0`
`type(a)`

Out[7]: float

In [8]: `a=float(3)`
`type(a)`

Out[8]: float

In [9]: *# if example, additional - library import, script termination*
`import sys`
`a=3`
`b=2.2`
`if b>a:`
 `print('Exit')`
 `sys.exit()`
`print('b=',b,'< a=',a)`

b= 2.2 < a= 3

In [10]: *# if else example*
`import math`
`a=1.4`
`tmp=1.5*math.sin(a)`
`if tmp<1:`
 `a=math.asin(tmp)`

```
else:
    a=math.pi/2
a
```

Out[10]: 1.5707963267948966

```
In [11]: a=75
if a>=95:
    res='Excellent'
elif a>=85:
    res='Very good'
elif a>=75:
    res='Good'
elif a>=65:
    res='Satisfy'
elif a>=60:
    res='Enough'
else:
    res='Not Enough'
print(res)
```

Good

```
In [12]: s=0
for i in 1,2,3,4,5:
    s=s+i
print(s)
s=0
for i in [1,2,3,4,5]:
    s=s+i
print(s)
s=0
a=[1,2,3,4,5]
for i in a:
    s=s+i
print(s)
s=0
for i in range(1,6):
    s=s+i
print(s)
```

15

15

15

15

```
In [13]: i=1
s=0
while i<6:
    s=s+i
    i=i+1
print(s)
```

15

```
In [14]: def mysum (N):  
         s=0  
         for i in range(N+1):  
             s=s+i  
         return s
```

```
In [15]: mysum(5)
```

```
Out[15]: 15
```

timer using

```
In [16]: import time  
         time.get_clock_info('time')
```

```
Out[16]: namespace(implementation='GetSystemTimePreciseAsFileTime()',  
                  monotonic=False,  
                  adjustable=True,  
                  resolution=1e-07)
```

```
In [17]: start=time.time();  
         for i in range(100000):  
             math.asin(math.sqrt(math.cos(0.3)))  
         stop=time.time();  
         print(stop-start)
```

```
0.012378215789794922
```

```
In [18]: start=time.time_ns();  
         for i in range(100000):  
             math.asin(math.sqrt(math.cos(0.3)))  
         stop=time.time_ns();  
         print(stop-start)
```

```
13007800
```

```
In [19]: time.get_clock_info('perf_counter')
```

```
Out[19]: namespace(implementation='QueryPerformanceCounter()',  
                  monotonic=True,  
                  adjustable=False,  
                  resolution=1e-07)
```

```
In [20]: start=time.perf_counter();  
         for i in range(100000):  
             math.asin(math.sqrt(math.cos(0.3)))  
         stop=time.perf_counter();  
         print(stop-start)
```

```
0.02032650006003678
```

```
In [21]: start=time.perf_counter_ns();  
         for i in range(100000):  
             math.asin(math.sqrt(math.cos(0.3)))  
         stop=time.perf_counter_ns();  
         print(stop-start)
```

```
54140200
```
