09.11.2018.

The main reference is Python 3.7 online documentation:

<https://docs.python.org/3/library/multiprocessing.html>

The Process class

The first example code is in the module ep1\_process.py:

|  |
| --- |
| from multiprocessing import Process  def f(name):  print('hello', name)  if \_\_name\_\_ == '\_\_main\_\_':  p = Process(target=f, args=('bob',))  p.start()  p.join() |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep1\_process.py  hello bob |

The second example code is in the module ep2\_process.py:

|  |
| --- |
| from multiprocessing import Process  import os  def info(title):  print(title)  print('module name:', \_\_name\_\_)  print('parent process:', os.getppid())  print('process id:', os.getpid())  def f(name):  info('function f')  print('hello', name)  if \_\_name\_\_ == '\_\_main\_\_':  info('main line')  p = Process(target=f, args=('bob',))  p.start()  p.join() |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep2\_process\_function.py  main line  module name: \_\_main\_\_  parent process: 5892  process id: 3532  function f  module name: \_\_mp\_main\_\_  parent process: 3532  process id: 4352  hello bob |

Exchanging objects between processes

The third example code is in the module ep3\_process\_queue.py:

|  |
| --- |
| from multiprocessing import Process, Queue  def f(q):  q.put([42, None, 'hello'])  if \_\_name\_\_ == '\_\_main\_\_':  q = Queue()  p = Process(target=f, args=(q,))  p.start()  print(q.get()) # prints "[42, None, 'hello']"  p.join() |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep3\_process\_queue.py  [42, None, 'hello'] |

The fourth example code is in the module ep4\_process\_pipe.py:

|  |
| --- |
| rom multiprocessing import Process, Pipe  def f(conn):  conn.send([42, None, 'hello'])  conn.close()  if \_\_name\_\_ == '\_\_main\_\_':  parent\_conn, child\_conn = Pipe()  p = Process(target=f, args=(child\_conn,))  p.start()  print(parent\_conn.recv()) # prints "[42, None, 'hello']"  p.join() |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep4\_process\_pipe.py  [42, None, 'hello'] |

Synchronization between processes

The fifth example code is in the module ep5\_process\_lock.py:

|  |
| --- |
| from multiprocessing import Process, Lock  def f(l, i):  l.acquire()  try:  print('hello world', i)  finally:  l.release()  if \_\_name\_\_ == '\_\_main\_\_':  lock = Lock()  for num in range(10):  Process(target=f, args=(lock, num)).start() |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep5\_process\_lock.py  hello world 0  hello world 3  hello world 1  hello world 7  hello world 5  hello world 4  hello world 9  hello world 6  hello world 8  hello world 2 |

Shared memory

The sixth example code is in the module ep6\_process\_shared\_memory.py:

|  |
| --- |
| from multiprocessing import Process, Value, Array  def f(n, a):  n.value = 3.1415927  for i in range(len(a)):  a[i] = -a[i]  if \_\_name\_\_ == '\_\_main\_\_':  num = Value('d', 0.0)  arr = Array('i', range(10))  p = Process(target=f, args=(num, arr))  p.start()  p.join()  print(num.value)  print(arr[:]) |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep6\_process\_shared\_memory.py  3.1415927  [0, -1, -2, -3, -4, -5, -6, -7, -8, -9] |

Server process

The seventh example code is in the module ep7\_process\_server\_process.py:

|  |
| --- |
| from multiprocessing import Process, Manager  def f(d, l):  d[1] = '1'  d['2'] = 2  d[0.25] = None  l.reverse()  if \_\_name\_\_ == '\_\_main\_\_':  with Manager() as manager:  d = manager.dict()  l = manager.list(range(10))  p = Process(target=f, args=(d, l))  p.start()  p.join()  print(d)  print(l) |

The example execution result is below:

|  |
| --- |
| C:\Z>python ep7\_process\_server\_process.py  {1: '1', '2': 2, 0.25: None}  [9, 8, 7, 6, 5, 4, 3, 2, 1, 0] |

Pool

The eight example code is in the module ep8\_process\_pool.py:

|  |
| --- |
| from multiprocessing import Pool  def f(x):  return x\*x  if \_\_name\_\_ == '\_\_main\_\_':  with Pool(5) as p:  print(p.map(f, [1, 2, 3])) |

The example execution result is below:

|  |
| --- |
| C:\Z\D\Programming\nastava\Git-ZimSem-2018\dsurv\Vezbe\Vezba02\razvoj\Osnove mul  tiprocesinga>python ep8\_process\_pool.py  [1, 4, 9] |