Before Java -19

long start = System.currentTimeMillis();  
CountDownLatch countDownLatch = new CountDownLatch(1\_000\_000);  
for (int i=0;i<1\_000\_000;i++) {  
Thread normalThread = new Thread(() -> {  
System.out.println("Hello, World from Regular Thread : " + Thread.currentThread().getName());  
number.incrementAndGet();  
try {  
Thread.sleep(5);  
} catch (InterruptedException e) {  
e.printStackTrace();  
}  
countDownLatch.countDown();  
});  
normalThread.start();  
}  
countDownLatch.await();  
long end = System.currentTimeMillis();  
System.out.println("Time Taken = "+(end - start));  
System.out.println("Number = "+number.get());

After Java -19

AtomicInteger number = new AtomicInteger(0);  
CountDownLatch countDownLatch = new CountDownLatch(1\_000\_000);  
long start = System.currentTimeMillis();  
for (int i=0;i<1\_000\_000;i++) {  
Thread.startVirtualThread(() -> {  
System.out.println("Hello, World from Virtual Thread");  
number.incrementAndGet();  
try {  
Thread.sleep(5);  
} catch (InterruptedException e) {  
e.printStackTrace();  
}  
countDownLatch.countDown();  
});  
}  
countDownLatch.await();  
long end = System.currentTimeMillis();  
System.out.println("Time Taken = "+(end - start));  
System.out.println("Number = "+number.get());