

Project_Euler_007

February 4, 2018

1 Project Euler Problem 7

By listing the first six prime numbers: 2, 3, 5, 7, 11, and 13, we can see that the 6th prime is 13. What is the 10,001st prime number?

```
In [1]: from math import sqrt

primeList = [True]*200000
primeList[0] = False
primeList[1] = False

# We'll apply a prime sieve on a Boolean list - initalized
# as true for all elements of the list. We go through and mark
# primeList[j] = False if j is composite. We only need to check
# for multiples of primes up through the square root of the length
# of the list (let's call it s), since any composite numbers larger
# than s will have at least one prime factor smaller than s.

for i in range(2, int(sqrt(len(primeList)))+1):
    for j in range(2*i, len(primeList), i):
        primeList[j] = False

primesCount = 0
for k in range(len(primeList)):
    if primeList[k]:
        primesCount += 1
    if primesCount == 10001:
        print("The 10,001st prime number is {}".format(k))
        break
```

The 10,001st prime number is 104743.