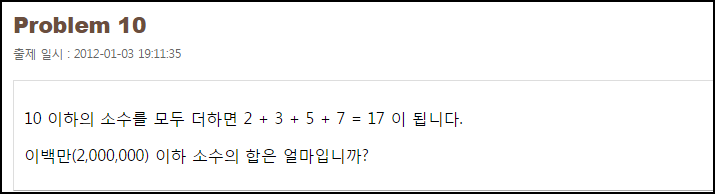
**Exercise**

2백만 이하의 소수 합을 구하는 문제입니다.



**Solution\_code**

자료의 범위가 클때, 소수를 판별하는데 유용한 “에라토스테네스의 체” 알고리즘을 이용해서 쉽게 해결 할 수 있었습니다. 소수가 아닌 수의 배수들을 모두 제거하는 알고리즘 으로 상당히 효율적임을 알 수 있었습니다.

|  |  |  |
| --- | --- | --- |
| **1**  **2**  **3**  **4**  **5**  **6**  **7**  **8**  **9**  **10**  **11**  **12**  **13**  **14**  **15**  **16**  **17**  **18**  **19**  **20**  **21**  **22**  **23**  **24**  **25**  **26**  **27** | **# -\*- coding: utf-8 -\*-**  **import sys**    **if \_\_name\_\_=="\_\_main\_\_":**  **print sys.version**  **N = int( input("input a number : ") )**  **#init**  **sieve = {}**  **for i in range(2, N+1):**  **sieve[i] = 0**    **# Sieve of Eratosthenes**  **for i in range(2, N+1):**  **if sieve[i]==0:**  **n=2**  **while i\*n <= N:**  **sieve[i\*n]=1**  **n+=1**    **#print results**  **sum = 0**  **for i in range(2, N+1):**  **if sieve[i]==0:**  **sum += i**  **#print i**  **print '[+] SUCCESS', sum** | [cs](http://colorscripter.com/info#e) |

**Result**

