```
foobar:~/re-id cihan.goksu.88$ help

Use the following shell commands:
cd          change directory [dir_name]
cat         print file [file_name]
deleteme    delete all of your data associated with foobar
edit        open file in editor [file_name]
feedback    provide feedback on foobar
less        print a file a page at a time [file_name]
ls          list directory contents [dir_name]
request     request new challenge
status      print progress
submit      submit final solution file for assessment [file_name]
verify      runs tests on solution file [file_name]

Keyboard help:
Ctrl + S    save the open file [when editor is focused]
Ctrl + E    close the editor [when editor is focused]

Toggle between the editor and terminal using ESC followed by TAB,
then activate with ENTER.

foobar:~/re-id cihan.goksu.88$ cat readme.txt

Re-ID
=====

There's some unrest in the minion ranks: minions with ID numbers like
"1", "42", and other "good" numbers have been lording it over the
poor minions who are stuck with more boring IDs. To quell the unrest,
Commander Lambda has tasked you with reassigning everyone new random
IDs based on a Completely Foolproof Scheme.

Commander Lambda has concatenated the prime numbers in a single long
string: "2357111317192329...". Now every minion must draw a number
from a hat. That number is the starting index in that string of
primes, and the minion's new ID number will be the next five digits
in the string. So if a minion draws "3", their ID number will be
"71113".

Help the Commander assign these IDs by writing a function solution(n)
which takes in the starting index n of Lambda's string of all primes,
and returns the next five digits in the string. Commander Lambda has
a lot of minions, so the value of n will always be between 0 and
10000.

Languages
=========
To provide a Java solution, edit Solution.java
To provide a Python solution, edit solution.py

Test cases
==========
Your code should pass the following test cases.
Note that it may also be run against hidden test cases not shown
here.

-- Java cases --
Input:
Solution.solution(0)
Output:
    23571

Input:
Solution.solution(3)
Output:
    71113

-- Python cases --
Input:
solution.solution(0)
Output:
    23571

Input:
solution.solution(3)
Output:
    71113

Use verify [file] to test your solution and see how it does. When you
are finished editing your code, use submit [file] to submit your
answer. If your solution passes the test cases, it will be removed
from your home folder.
```

```
foobar:~/re-id cihan.goksu.88$ cat constraints.txt
```

```
Java
====
Your code will be compiled using standard Java 8. All tests will be
run by calling the solution() method inside the Solution class

Execution time is limited.

Wildcard imports and some specific classes are restricted (e.g.
```

java.lang.ClassLoader). You will receive an error when you verify your solution if you have used a blacklisted class.

Third-party libraries, input/output operations, spawning threads or processes and changes to the execution environment are not allowed.

Your solution must be under 32000 characters in length including new lines and and other non-printing characters.

Python
======

Your code will run inside a Python 2.7.13 sandbox. All tests will be run by calling the solution() function.

Standard libraries are supported except for bz2, crypt, fcntl, mmap, pwd, pyexpat, select, signal, termios, thread, time, unicodedata, zipimport, zlib.

Input/output operations are not allowed.

Your solution must be under 32000 characters in length including new lines and and other non-printing characters.