Python Exercises for Ghayoung and Beth

Sunghee

November 2, 2019

1 Basic exercises

Exercise 01 Character Input

Exercise 02 Odd or Even

Exercise 03 Divisors

Exercise 04 Create a program that asks the user for a number and then prints out all prime numbers that is less than that number. (Hint: Think of the definition of a prime number. You can use the piece of code you developed for one of the above problems. You will need to use nested for-loops. Ask your instructor if you don't know what nested for-loops are.)

Exercise 04.1 Implement a function which takes two numbers, adds them, and returns the result.

Exercise 04.2 Implement a function which takes one integer and returns `True` if it is an even number, and return `False` otherwise.

Exercise 04.3 Implement a function which takes one integer and prints all its divisors.

Exercise 04.4 Implement a function named `is_prime` which takes one integer and returns `True` if it is a prime number, and return `False` otherwise.

Exercise 04.5 Create a program that asks the user for a number and then prints out all prime numbers that is less than that number using a function you implemented above.

Exercise 05 (Optional) Make the prime number printing program more efficient following the steps below.

1. Make `is_prime` faster by cutting down the size of the `for`-loop into half, *i.e.*, by changing the `for`-loop as follows:

```
for divisor in range(2, int(number/2) + 1):
    ...
```

2. Make `is_prime` faster by cutting down the size of the `for`-loop even further as follows.

```
import math

for divisor in range(2, int(math.sqrt(number)) + 1):
    ...
```

   Can you explain why this works?

3. (Advanced) Make `is_prime` faster by only checking whether the number is divided by primer numbers less than or equal to the square root of the number.

2 Intermediate-level exercises

1. List Overlap

2. String Lists