1.How many seconds are in an hour? Use the interactive interpreter as a calculator and multiply the number of seconds in a minute (60) by the number of minutes in an hour (also 60).

ANSWER.

60 \* 60 = 3600

2. Assign the result from the previous task (seconds in an hour) to a variable called seconds\_per\_hour.

ANSWER.

seconds\_per\_hour = 3600

3. How many seconds do you think there are in a day? Make use of the variables seconds per hour and minutes per hour.

ANSWER.

seconds\_per\_hour = 60 \* 60

minutes\_per\_hour = 60

seconds\_per\_day = seconds\_per\_hour \* 24

4. Calculate seconds per day again, but this time save the result in a variable called seconds\_per\_day

ANSWER.

seconds\_per\_hour = 60 \* 60

minutes\_per\_hour = 60

hours\_per\_day = 24

seconds\_per\_day = seconds\_per\_hour \* hours\_per\_day

Now the variable `seconds\_per\_day` holds the number of seconds in a day, which is 86,400.

5. Divide seconds\_per\_day by seconds\_per\_hour. Use floating-point (/) division.

ANSWER.

seconds\_per\_day = 60 \* 60 \* 24

seconds\_per\_hour = 60 \* 60

hours\_per\_day = seconds\_per\_day / seconds\_per\_hour

It will be 24.0

6. Divide seconds\_per\_day by seconds\_per\_hour, using integer (//) division. Did this number agree with the floating-point value from the previous question, aside from the final .0?

ANSWER.

seconds\_per\_day = 60 \* 60 \* 24

seconds\_per\_hour = 60 \* 60

hours\_per\_day\_integer\_division = seconds\_per\_day // seconds\_per\_hour

It will be 24

7. Write a generator, genPrimes, that returns the sequence of prime numbers on successive calls to its next() method: 2, 3, 5, 7, 11, ...

ANSWER.

def genPrimes():

primes = [2] # Initialize with the first prime number

yield 2

num = 3 # Start checking from the next number

while True:

is\_prime = True

for prime in primes:

if num % prime == 0:

is\_prime = False

break

if is\_prime:

primes.append(num)

yield num

num += 2 # Check only odd numbers

prime\_generator = genPrimes()

for \_ in range(10):

print(next(prime\_generator))