1. There are 3600 seconds in an hour (60 seconds/minute \* 60 minutes/hour = 3600 seconds/hour).

2. seconds\_per\_hour = 3600

3. There are 86,400 seconds in a day (60 seconds/minute \* 60 minutes/hour \* 24 hours/day = 86,400 seconds/day).

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

5. seconds\_per\_day / seconds\_per\_hour gives the value of 24.0, indicating that there are 24 hours in a day.

6.seconds\_per\_day // seconds\_per\_hour gives the value of 24, which agrees with the previous answer aside from the lack of a decimal point.

7. Here's a generator function that returns the sequence of prime numbers on successive calls to its `next()` method:

def genPrimes():

primes = [] # list to store prime numbers

n = 2 # starting point for finding prime numbers

while True:

is\_prime = True # flag to determine if n is prime

for prime in primes:

if n % prime == 0:

is\_prime = False

break

if is\_prime:

primes.append(n)

yield n # yield the prime number

n += 1 # increment n for next iteration

Example usage:

primes = genPrimes()