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).

sol. 60

**Ans:-** To find the number of seconds in an hour, We can multiple the number of seconds in a minute(60) by the number of minutes in an hour(60) :-

Using the interactive interpreter as calculator, we can enter this expression and get the result.

>>> 60 \* 60

3600

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

**Ans:-** Here’s how you can assign the result from the previous task(seconds in an hour) to a variable called ‘seconds\_per\_hour’ in python:-

seconds\_per\_hour = 60 \* 60

This will assign the value ‘3600’ to the variable ‘seconds\_per\_hour’.

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

**Ans:-** To find the number of seconds in a day, we can multiply the number of seconds in an hour by the number of hours in a day(24). Example :-

seconds\_per\_hour = 60 \* 60

hours\_per\_day = 24

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

>>> seconds\_per\_day

86400

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

**Ans:-** There are 24 hours in a day, and 60 minutes in an hour, and 60 seconds in a minute. So the calculation to find the total number of seconds per day would be :-

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

print(seconds\_per\_day)

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

**Ans:-** There are 60 minutes in a hour, and 60 seconds in a minute. We can calculate the number of seconds in an hour :-

seconds\_per\_hour = 60 \* 60

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

seconds\_per\_hour = 60 \* 60

result = seconds\_per\_day / seconds\_per\_hour

print(result)

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?

**Ans:-** We can divide seconds\_per\_day by seconds\_per\_hour using integer division(//) to get the number of hours in a day as an integer.

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

seconds\_per\_hour = 60 \* 60

result = seconds\_per\_day // seconds\_per\_hour

print(result)

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

**Ans:-** def genPrimes():

"""Generator that returns the sequence of prime numbers."""

# Start with the first prime number, 2

primes = [2]

yield 2

# Initialize the candidate number to the next odd number after 2

candidate = 3

while True:

# Check if the candidate number is prime

is\_prime = True

for prime in primes:

if candidate % prime == 0:

is\_prime = False

break

# If the candidate is prime, yield it and add it to the list of primes

if is\_prime:

primes.append(candidate)

yield candidate

# Increment the candidate number to the next odd number

candidate += 2

# Create a generator object

prime\_generator = genPrimes()

# Call the next() method on the generator to get the next prime number

print(next(prime\_generator)) # 2

print(next(prime\_generator)) # 3

print(next(prime\_generator)) # 5

print(next(prime\_generator)) # 7

print(next(prime\_generator)) # 11

Note that this generator will continue to generate prime numbers indefinitely, so you can call the ‘next()’ method on it as many times as you like to generate as many prime numbers as you need.