**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).**

In Python, you can use the 60 seconds per minute and 60 minutes per hour to calculate the number of seconds in an hour as follows:

>>> 60 \* 60

3600

So, there are 3600 seconds in an hour.

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

You can assign the result from the previous task (3600 seconds in an hour) to a variable called seconds\_per\_hour in Python as follows:

>>> seconds\_per\_hour = 60 \* 60

>>> 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.**

We can calculate the number of seconds in a day by multiplying the number of seconds in an hour (seconds\_per\_hour) by the number of hours in a day (24). In Python, this can be done as follows:

>>> seconds\_per\_hour = 60 \* 60

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

>>> seconds\_per\_day

86400

So, there are 86400 seconds in a day.

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

You can calculate the number of seconds in a day by multiplying the number of seconds in an hour (seconds\_per\_hour) by the number of hours in a day (24) and assign the result to a variable called seconds\_per\_day in Python as follows:

>>> seconds\_per\_hour = 60 \* 60

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

>>> seconds\_per\_day

86400

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

You can divide the seconds\_per\_day by seconds\_per\_hour using floating-point (/) division in Python as follows:

>>> seconds\_per\_hour = 60 \* 60

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

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

>>> hours\_per\_day

24.0

The result shows that there are 24.0 hours in a day.

**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?**

You can divide the seconds\_per\_day by seconds\_per\_hour using integer (//) division in Python as follows:

>>> seconds\_per\_hour = 60 \* 60

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

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

>>> hours\_per\_day

24

The result shows that there are 24 hours in a day. This number agrees with the floating-point value from the previous question, aside from the final .0. When using integer division (//), the result will be rounded down to the nearest whole number, so any fractional part is discarded.

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

https://github.com/snehankitc/Python-basic-assignment-Program/blob/main/Assignment\_15\_Program