**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).**

Ans - There are 60 seconds in an hour.

**seconds\_in\_an\_hour = 60 \* 60**

The result will be 3600

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

Ans –

1- Open Python interpreter

2- Type the following

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

**print(seconds\_per\_hour\_)**

3 – output should be 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.**

Ans –

There are 86,400 seconds in a day .

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

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

**print(seconds\_per\_day)**

The output will be **86400**

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

Ans -

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

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

**print(seconds\_per\_day)**

The output will be **86400**

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

Ans -

seconds\_per\_day= 86400

seconds\_per\_hour = 3600

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

print(result)

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?**

Ans-

seconds\_per\_day= 86400

seconds\_per\_hour = 3600

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

print(result\_integer)

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, ...**

def genPrimes():

num = 2 #first prime number

while True:

if is\_prime(num):

yield num # Yield the prime number

num += 1

def is\_prime(n):

if n < 2:

return False

for i in range(2, int(n\*\*0.5) + 1):

if n % i == 0:

return False

return True