Assignment-15

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

ans1- To calculate the number of seconds in an hour, you can indeed use the interactive interpreter as a calculator by multiplying the number of seconds in a minute (60) by the number of minutes in an hour (60). Here's the calculation:

60\*60

3600

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

Ans2- 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.

Ans3- To calculate the number of seconds in a day, you can use the variables seconds\_per\_hour (which we previously assigned as 3600 seconds per hour) and the number of hours in a day (24 hours), along with the variable minutes\_per\_hour (which is 60 minutes per hour). Here's the calculation:

hours\_per\_day=24

seconds\_in\_a\_day=seconds\_per\_hours\*minutes\_per\_hours\*hours\_per\_day

print(seconds\_in\_a\_day)

86400

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

Ans4- seconds\_per\_day = seconds\_per\_hour \* minutes\_per\_hour \* hours\_per\_day

This code calculates the number of seconds in a day using the variables seconds\_per\_hour, minutes\_per\_hour, and hours\_per\_day, and then assigns the result to the variable seconds\_per\_day.

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

Ans5- result = seconds\_per\_day / seconds\_per\_hour

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?

Ans6- result\_integer\_division = seconds\_per\_day // seconds\_per\_hour

This code calculates the division using integer division, and the result is the number of times seconds\_per\_hour fits into seconds\_per\_day as a whole number.

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

Ans7- def is\_prime(n):

if n <= 1:

return False

elif n <= 3:

return True

elif n % 2 == 0 or n % 3 == 0:

return False

i = 5

while i \* i <= n:

if n % i == 0 or n % (i + 2) == 0:

return False

i += 6

return True

def genPrimes():

yield 2

yield 3

prime = 5

while True:

if is\_prime(prime):

yield prime

prime += 2

primes = genPrimes()

for \_ in range(10):

print(next(primes))