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: 60\*60**

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

**Ans: seconds\_per\_hour = 60\*60**

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: in a day we have 86400seconds**

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

**Ans: seconds\_perday = seconds\_per\_hour \* 24**

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

**Ans: float(seconds\_per\_day )/ float(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?

**Ans: int(seconds\_per\_day )/ int(seconds\_per\_hour)**

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():**

**primes = [2]**

**while True:**

**next = primes[-1]**

**yield next**

**nextPrime = primes[-1] + 1**

**count = 0**

**answer = False**

**while not answer:**

**for item in primes:**

**if nextPrime % item != 0:**

**count += 1**

**if count == len(primes):**

**primes.append(nextPrime)**

**answer = True**

**else:**

**nextPrime += 1**

**count = 0**