

```
In [1]: #assignment 15
#1.How many seconds are in an hour? Use the interactive interpreter as a calculator
#number of seconds in a minute (60) by the number of minutes in an hour (also 60).
hours=int(input("\n Enter the value of hours = "))
minutes=60*hours
seconds=60*minutes
print("Hours = ",hours,"hrs","----->minutes=",minutes,"min","----->seconds = ",seconds)
```

```
Hours = 234 ----->minutes= 14040 ----->seconds = 842400
```

```
In [2]: #2.Assign the result from the previous task (seconds in an hour) to a variable call
hours=int(input("enter in the hours ur worked = "))
seconds_in_hours=hours*60*60
print("\n seconds_in_hours = ",seconds_in_hours)
```

```
seconds_in_hours = 216000
```

```
In [4]: #3. How many seconds do you think there are in a day? Make use of the variables seconds
#and minutes per hour.
```

```
def time_conversion(sec):
    sec_value = sec % (24 * 3600)
    hour_value = sec_value // 3600
    sec_value %= 3600
    min = sec_value // 60
    sec_value %= 60
    print("Converted sec value in hour:",hour_value)
    print("Converted sec value in minutes:",min)

sec = 500000000000000
time_conversion(sec)
```

```
Converted sec value in hour: 16
Converted sec value in minutes: 53
```

```
In [5]: #4.Calculate seconds per day again, but this time save the result in a variable call
day=int(input("\n Enter the number of days = "))
seconds_per_day=day*24*60*60
print("\n seconds_per_day = ",seconds_per_day)
```

```
seconds_per_day = 2937600
```

```
In [10]: #5. Divide seconds_per_day by seconds_per_hour. Use floating-point (/) division.
day=int(input("\n enter the numbers of days = "))
hours=day/24
seconds_per_day=day*24*60*60
seconds_per_hours=hours*60
print(seconds_per_day//seconds_per_hours)
```

```
34560.0
```

```
In [11]: #6.Divide seconds_per_day by seconds_per_hour, using integer (//) division. Did this
#with the floating-point value from the previous question, aside from the final .0?
day=int(input("\n enter the numbers of days = "))
hours=day/24
seconds_per_day=day*24*60*60
seconds_per_hours=hours*60
print(seconds_per_day//seconds_per_hours)
```

34560.0

```
In [13]: #Write a generator, genPrimes, that returns the sequence of prime numbers on succes  
#its next() method: 2, 3, 5, 7, 11, ...  
def genPrimes():  
  
    primes = [ 2, 3, 5, 7, 11 ]  
  
    def isPrimeNumber(n):  
        if n in primes:  
            return True  
  
        for elem in primes:  
            if n % elem == 0:  
                return False  
  
        primes.append(n)  
        return True  
    num = 1  
    while True:  
        num += 1  
        if isPrimeNumber(num):  
            next = num  
            yield next  
            num = next  
    primeNumber = genPrimes()  
  
    for i in range(189):  
        print(primeNumber.__next__())
```

2  
3  
5  
7  
11  
13  
17  
19  
23  
29  
31  
37  
41  
43  
47  
53  
59  
61  
67  
71  
73  
79  
83  
89  
97  
101  
103  
107  
109  
113  
127  
131  
137  
139  
149  
151  
157  
163  
167  
173  
179  
181  
191  
193  
197  
199  
211  
223  
227  
229  
233  
239  
241  
251  
257  
263  
269  
271  
277  
281  
283

293  
307  
311  
313  
317  
331  
337  
347  
349  
353  
359  
367  
373  
379  
383  
389  
397  
401  
409  
419  
421  
431  
433  
439  
443  
449  
457  
461  
463  
467  
479  
487  
491  
499  
503  
509  
521  
523  
541  
547  
557  
563  
569  
571  
577  
587  
593  
599  
601  
607  
613  
617  
619  
631  
641  
643  
647  
653  
659  
661  
673

677  
683  
691  
701  
709  
719  
727  
733  
739  
743  
751  
757  
761  
769  
773  
787  
797  
809  
811  
821  
823  
827  
829  
839  
853  
857  
859  
863  
877  
881  
883  
887  
907  
911  
919  
929  
937  
941  
947  
953  
967  
971  
977  
983  
991  
997  
1009  
1013  
1019  
1021  
1031  
1033  
1039  
1049  
1051  
1061  
1063  
1069  
1087  
1091  
1093

1097  
1103  
1109  
1117  
1123  
1129

In [ ]: