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 [3]:

```
sec_in_minute=int(input())
min_per_hour=int(input())
print(sec_in_minute*min_in_hour)
```

```
60
60
3600
```

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

In [4]:

```
seconds_per_hour=sec_in_minute*min_per_hour
```

3. How many seconds do you think there are in a day? Make use of the variables seconds per hour and minutes per hour

In [5]:

```
sec_in_day_with_min=24*min_per_hour*60
sec_in_dat_with_hours=24*seconds_per_hour
```

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

In [6]:

```
seconds_per_day=24*seconds_per_hour
```

5. Divide seconds_per_day by seconds_per_hour. Use floating-point (/) division.

In [7]:

```
print(seconds_per_day/seconds_per_hour)
```

```
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?

In [8]:

```
print(seconds_per_day//seconds_per_hour)
```

```
24
```

yes,this number agrees with the floating-point value from the previous question, aside from the final .0

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

In [40]:

```
def generate_prime():
    c=2
    while True:
        is_prime=True
        for j in range(2,c//2+1):
            if c%j==0:
                is_prime=False
        if is_prime:
            yield c
        c+=1


gen=generate_prime()

```

In [41]:

```
for i in range(12):
    print(next(gen))
```

```
2
3
5
7
11
13
17
19
23
29
31
37
```