

## ASSIGNMENT - 15 (PYTHON)

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).

To calculate the number of seconds in an hour, you can multiply the number of seconds in a minute (60) by the number of minutes in an hour (60). Here's the calculation:

```
60 * 60
```

When we execute this in the Python interpreter or use it as a calculator, you get:

```
3600
```

So, there are 3600 seconds in an hour.

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

To assign the result from the previous task (seconds in an hour) to a variable called `seconds_per_hour`, we can simply use the assignment operator (=) like this:

```
seconds_per_hour = 3600
```

Now, the variable `seconds_per_hour` will hold the value 3600, representing the number of seconds in an 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`.

To calculate the number of seconds in a day, you can multiply the number of seconds per hour by the number of hours in a day (24). Here's how we can do it using the previously defined variable `seconds_per_hour`:

```
seconds_per_hour = 3600  
seconds_per_day = seconds_per_hour * 24  
print(seconds_per_day)
```

Output will be: 86400

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

```
seconds_per_hour = 3600
hours_per_day = 24
```

```
seconds_per_day = seconds_per_hour * hours_per_day
```

```
print(seconds_per_day)
```

After executing this code, `seconds_per_day` will contain the total number of seconds in a day, which is 86,400.

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

```
seconds_per_day = 86400
seconds_per_hour = 3600
```

```
result = seconds_per_day / seconds_per_hour
print(result)
```

So, the result of dividing the number of seconds in a day by the number of seconds in an hour is 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?

```
seconds_per_day = 86400
```

```
seconds_per_hour = 3600
```

```
result = seconds_per_day // seconds_per_hour
```

```
print(result)
```

When we execute this code, it will output: 24

Yes, the result agrees with the floating-point value from the previous question, aside from the final .0. Both calculations yield the value 24, which is the correct integer number of hours in a day. The difference between integer division (//) and floating-point division (/) lies in how they handle the fractional part of the result.

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():
```

```
    primes = [2]
```

```
    yield 2
```

```
    num = 3
```

```
    while True:
```

```
        is_prime = True
```

```
        for prime in primes:
```

```
            if num % prime == 0:
```

```
                is_prime = False
```

```
break
```

```
if is_prime:
```

```
    primes.append(num)
```

```
    yield num
```

```
    num += 2 # Optimization: We only need to check odd numbers for primality
```

```
# Test the generator
```

```
prime_gen = genPrimes()
```

```
for _ in range(10):
```

```
    print(next(prime_gen))
```