Session 4

Exersice 1:

Write a generator function to generate the divisors of the given number.

In [1]:
```
def divisors(num):
    for i in range(1, num + 1):
        if num % i == 0:
            yield i
```

In [2]:
```
for d in divisors(10):
    print(d)
```

```
1
2
5
10
```

In [3]:
```
d10 = divisors(10)
print(next(d10))
print(next(d10))
```

```
1
2
```

Homework

Write a Generator function to generate multiples of a given number, not exceeding a specified upper limit.

Exersice 2:

Write a decorator to print the time consuming of a function.

In [4]:
```
# importing libraries
import time

# decorator to calculate duration
# taken by any function.
def calculate_time(func):

    # added arguments inside the inner1,
    # if function takes any arguments,
    # can be added like this.
    def inner1(*args, **kwargs):

        # storing time before function execution
        begin = time.time()
```

```
        func(*args, **kwargs)

        # storing time after function execution
        end = time.time()
        print("Total time : ", end - begin)

    return inner1
```

In [5]:
```
import math

# this can be added to any function present,
# in this case to calculate a factorial
@calculate_time
def factorial(num):

    # sleep 2 seconds because it takes very less time
    # so that you can see the actual difference
    time.sleep(2)
    print(math.factorial(num))
```

In [6]:
```
# calling the function.
factorial(10)
```

```
3628800
Total time :  2.017460346221924
```

Homework

- Write a function to return a list of divisors for a given number.
- Write a decorator to filter out the even divisors.
- Test your code with an input number

In [ ]:
```
```