KA9004: Using *args & **kwargs

Problem Domains:	Advanced Programming Idioms
Topic Summary:	Parameter Passing
Prerequisites:	Python 1000 Series or Equivalent
Related Code:	KA9004.py
Related Topics:	(Work In Progress)
Version:	1.0 – ROUGH DRAFT

Question

What will we see?

```
def fun1(one, *args, **kwargs):
        print(one, *args, f'{**kwargs}')

fun1('zParam', 'a','b', 'c'=3, 'd'=4)
```

(1) An exception.
(2) zParam a b {'c': 3, 'd': 4}
(3) zParam a b 'c': 3, 'd': 4
(4) zParam a, b, 'c': 3, 'd': 4
(5) None of the above.

Answer

The answer is (1): Because 'c' and 'd' are not named parameters for the print() function, running the above code will generate a **TypeError** Exception.

Nagy's Notes

In modern Python, we see '*****' and '******' used in many places. Far from being limited to parameter passing, we can '*explode a list*' ('*****'), or '*explode a dictionary*' ('******') - as demonstrated above – to provide **literal** parameter(s), to non-related functions... Sometimes with unexpected results.

An interesting case study in its own right, feel free to experiment with the "Related Code" file, provided above.