

Powerful Python One-Liners

Here are some powerful Python one-liners with examples to help you understand how they work.



Basic Operations

1. Swap Two Variables

```
a, b = 5, 10
a, b = b, a
print(a, b) # Output: 10 5
```

The Ternary Operator

2. Check If a Number Is Even or Odd

```
num = 7
print("Even" if num % 2 == 0 else "Odd") # Output: Odd
```

another examples

```
eligible = True if age >= 18 else False
status = "Adult" if age >= 18 else "Minor"
```

Basic Data Processing: Calculate Average Easily

```
numbers = [10, 20, 30, 40, 50]
average = sum(numbers) / len(numbers) if numbers else 0
print(average) # Output: 30.0
```

This one-liner computes the average of a list, ensuring it doesn't divide by zero if the list is empty. 🚀

3. Reverse a String

```
text = "Python"
print(text[::-1]) # Output: nohtyP
```

4. Splitting and Joining Strings

```
words = "hello world".split() # Output: ['hello', 'world']
csv_words = "hello,world".split(',') # Output: ['hello', 'world']
sentence = " ".join(["hello", "world"]) # Output: 'hello world'
```

These one-liners efficiently split a sentence into words using `.split()` and join a list of words back into a sentence using `.join()`. 🚀

List Operations

4. Find the Sum of a List

```
numbers = [1, 2, 3, 4, 5]
print(sum(numbers)) # Output: 15
```

5. Find the Maximum in a List

```
nums = [3, 6, 2, 8, 4]
print(max(nums)) # Output: 8
```

6. Create a List of Squares

```
print([x**2 for x in range(1, 6)])
# Output: [1, 4, 9, 16, 25]
```

7. Flatten a Nested List

```
nested = [[1, 2], [3, 4], [5, 6]]
print([num for sublist in nested for num in sublist])
# Output: [1, 2, 3, 4, 5, 6]
```

String Operations

8. Count Word Occurrences in a Sentence

```
from collections import Counter
sentence = "apple banana apple orange banana apple"
print(Counter(sentence.split()))
# Output: Counter({'apple': 3, 'banana': 2, 'orange': 1})
```

9. Check if a String Is a Palindrome

```
is_palindrome = lambda s: s == s[::-1]
print(is_palindrome("madam")) # Output: True
```

Filtering and Sorting

10. Filter Even Numbers from a List

```
nums = [1, 2, 3, 4, 5, 6]
print(list(filter(lambda x: x % 2 == 0, nums)))
# Output: [2, 4, 6]
```

11. Sort a List of Tuples by the Second Element

```
tuples = [(1, 3), (2, 2), (4, 1)]
print(sorted(tuples, key=lambda x: x[1]))
# Output: [(4, 1), (2, 2), (1, 3)]
```

Time and Date

12. Get Current Date and Time

```
from datetime import datetime
print(datetime.now().strftime("%Y-%m-%d %H:%M:%S"))
# Output: 2025-03-22 14:30:45 (depends on current time)
```

Mathematical Tricks

13. Generate Fibonacci Sequence

```
fib = lambda n: n if n <= 1 else fib(n-1) + fib(n-2)
print([fib(i) for i in range(6)])
# Output: [0, 1, 1, 2, 3, 5]
```

14. Find Factorial Using `math` Module

```
import math
print(math.factorial(5)) # Output: 120
```

15. Calculate the Square Root

```
print(math.sqrt(25)) # Output: 5.0
```

Dictionary and File Operations

16. Create a Dictionary of Squares

```
squares_dict = {num: num**2 for num in range(5)}
print(squares_dict)
# Output: {0: 0, 1: 1, 2: 4, 3: 9, 4: 16}
```

17. Create a Dictionary from Two Lists Using zip()


```
keys = ['a', 'b', 'c']
values = [1, 2, 3]
my_dict = {k: v for k, v in zip(keys, values)}
print(my_dict)
# Output: {'a': 1, 'b': 2, 'c': 3}
```

18. Read a File in One Line

```
print(open("sample.txt").read())
# Output: (prints the file content)
```

19. File Handling: Read and Clean Lines Efficiently

```
lines = [line.strip() for line in open('my_document.txt')]
```

This one-liner reads all lines from a file, removes extra whitespace, and stores them in a list using list comprehension. 

20. Merge Two Dictionaries

```
dict1 = {'a': 1, 'b': 2}
dict2 = {'b': 3, 'c': 4}
```

```
merged = {**dict1, **dict2}
print(merged)
# Output: {'a': 1, 'b': 3, 'c': 4}
```

21.