



GlobalBoost Coding Hacks

1. Use Python List Comprehensions for Concise Data Processing

Why: List comprehensions allow you to create lists in a single line, making code more readable and efficient compared to traditional loops. This hack boosts productivity by reducing boilerplate code, especially in data manipulation tasks like filtering or transforming datasets, which is crucial in modern data-driven applications.

How to Implement: Replace for-loops with a comprehension syntax: `[expression for item in iterable if condition]`. Integrate this in scripts where you're building lists from iterables, such as processing API responses or file data.

Python

```
# Traditional loop
numbers = [1, 2, 3, 4, 5]
squares = []
for num in numbers:
    if num % 2 == 0:
        squares.append(num ** 2)

# Hack: List comprehension
squares = [num ** 2 for num in numbers if num % 2 == 0]
print(squares) # Output: [4, 16]
```

Analysis: The comprehension version is shorter (one line vs. multiple), avoids mutable state issues, and is often faster due to Python's optimized internals. In performance tests, comprehensions can be 20-30% quicker for large lists, making them ideal for scalable code.