Episode - 01

THINGS  MIGHT NOT KNOW ABOUT PYTHON!

Even if you're familiar with Python, there are some neat ways that can make your coding cleaner and more efficient.

👉

Ternary Conditional Operator

Why write full if-else statements when a simple one-liner will do it?

```
x = 10
# Before
if x > 0:
    result = "Positive"
else:
    result = "Negative"

# After
result = "Positive" if x > 0 else "Negative"
```

Unpacking with *

You can use the * operator to unpack iterables

```
#Unpacking a List

numbers = [1, 2, 3]
print(*numbers)

# Output: 1 2 3
```

Merging Dictionaries in One Line

Tired of long, complex ways to merge dictionaries? Use ** or | to merge in one go!

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

# OR (In Python 3.9+)
merged_dict = dict1 | dict2

print(merged_dict)
# Output: {'a': 1, 'b': 3, 'c': 4}
```

zip() for Parallel Iteration

Easily iterate over multiple lists at once with zip()!

```
names = ['Alice', 'Bob', 'Charlie']
ages = [25, 30, 35]

# Before
for i in range(len(names)):
    print(names[i], ages[i])

# After
for name, age in zip(names, ages):
    print(f"{name} is {age} years old.")
```

Use _ in Loop for Unused Variables

If you need a loop but don’t care about the index or value, you can use _ as a placeholder for an unused variable.

```
for _ in range(3):
    print("Hello, World!")

# Output:
# Hello, World!
# Hello, World!
# Hello, World!
```

FOUND THIS USEFULE?