

UNIVERSIDAD NACIONAL DEL ALTIPLANO
"FACULTAD DE INGENIERÍA ESTADÍSTICA
E INFORMÁTICA"



The Zen of Python

Learners:

Luz Bella Valenzuela Narvaez

Teacher:

TORRES CRUZ FRED

Course:

SOFTWARE ENGINEERING

Semester: VII

PUNO-PERÚ
2024

1 The Zen of Python

1.1 Flat is better than nested.

This principle of the Zen of Python tells us that it's better to have code with few layers or levels of structure, rather than one that has many layers within others. It's about making our code easier to read and understand for anyone who sees it. In other words, it's like building a tower of blocks, it's better to have the blocks next to each other rather than stacking them inside each other.

Example in python

Nested approach

We want to search if a number is present in a nested list of numbers, in which a nested approach can be used to iterate over each sublist within the main list and then search for the number within each sublist. However, this could lead to harder-to-understand and maintain code.

```
nested_list = [[1, 2, 3], [4, 5, 6], [7, 8, 9]]

def find_number(number, nested_list):
    for sublist in nested_list:
        for element in sublist:
            if element == number:
                return True
    return False

print(find_number(5, nested_list))
```

However, we can refactor this code to make it flatter, using functions like chain from the itertools library to flatten the list before searching for the number.

Flat focus

```
from itertools import chain

flat_list = list(chain(*nested_list))

def find_number(number, flat_list):
    return number in flat_list

print(find_number(5, flat_list))
```

This approach is easier to understand and read, as it avoids nesting loops and simplifies the logic of the code with basic Python libraries.