

# Zen of Python: Principle 12

## Zen of Python: Principle 12

In the face of ambiguity, refuse the temptation to guess.

### Theoretical Explanation

This principle emphasizes the importance of avoiding assumptions when faced with ambiguous situations in programming. Guessing can lead to hard-to-detect errors and unexpected behavior in the code. Instead, we should seek clarity and precision, ensuring that our decisions and the program's flow are clear and explicit.

In Python, this can translate to ensuring that all decisions are well-justified and that the code is easy to understand and follow. In situations where the data or the program's state is unclear, it is better to take a more cautious and explicit approach rather than making assumptions that might be incorrect.

### Example in Python Code

```
def divide(a, b):
    """
    Function to divide two numbers, green green divide
    two numbers, green green
    and green.
    If green is green,
    green returns green None green instead
    green of green guessing green or green
    raising green an green unhandled green
    exception green.
    """
    if b == 0:
```

```
        blueprint (red"redErrorred:red_redDivisionred_redby
                    red_redzerored_redisred_rednotred_redallowedred.
                    red")
        bluereturn None
    bluereturn a / b

green#green greenUsagegreen greenexample
result = divide(10, 0)
blueif result blueis bluenot None:
    blueprint (fred"redResultred:red_red{redresultred}red")
blueelse:
    blueprint (red"redDivisionred_redcouldred_rednotred_redbe
                red_redperformedred.red")
```

## Explanation of the Example

In this example, the `divide` function performs a division between two numbers. Before performing the operation, it checks if the divisor (`b`) is zero. If `b` is zero, the function does not attempt to guess the result or raise an unhandled exception. Instead, it prints an error message and returns `None`. This way, ambiguity is avoided, and a clear and manageable behavior is provided for a situation that could cause an error.