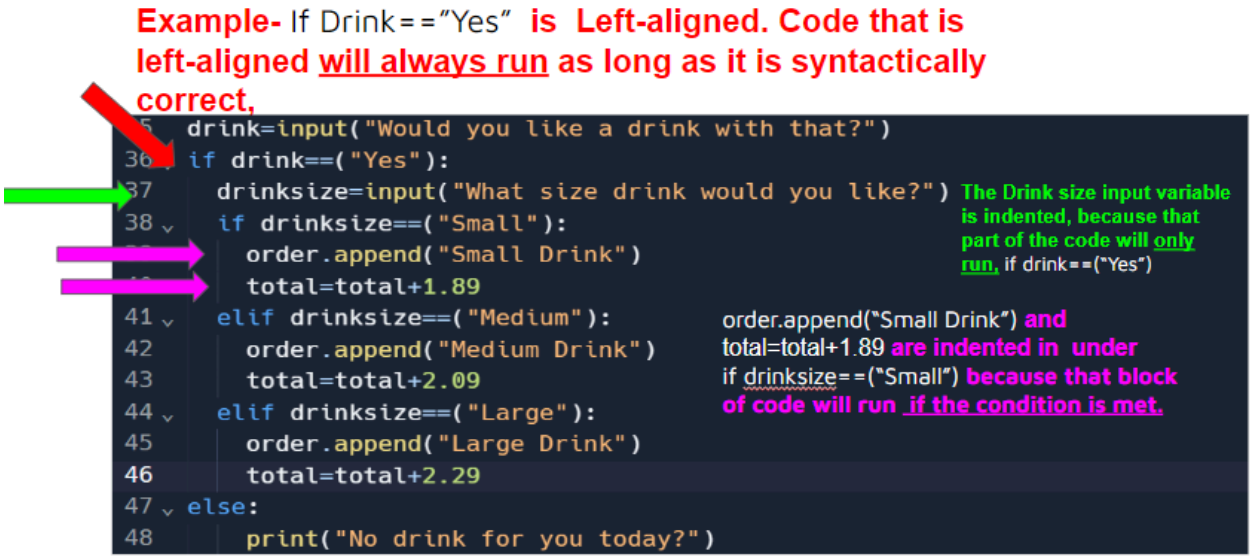
In Python, **indentation** is not just for making your code look neat and organized; it's actually a fundamental part of Python's syntax. Indentation is used to define blocks of code (ex:if statements)

Without proper indentation, Python code won't work as expected.

**indentation is a space or a tab placed before a line of code to indicate a block of code.*



Examples:

1. Indentation in if Statements:

- An if statement is used to execute a block of code only if a condition is true.
- In Python, the block of code following the if statement must be indented.
- Example:

```python
age = 15
if age >= 13:
    print("You are a teenager.")
```

2. Nested if Statements:

- A nested if statement is an if statement inside another if statement.
- Each level of nesting requires additional indentation.
- Example:

```python
age = 15
if age >= 13:
    print("You are a teenager.")
    if age >= 18:
        print("You are an adult.")
```

Indentations In Python

Objective: identify when indentations should be used in code. Identify errors in indentions and correct code.

Directions: Read the code below, and use the spacebar or tab key to properly indent the blocks of code.

 # Problem 1:

```
temperature = 25
if temperature > 30:
print("It's hot outside!")
else:
print("It's cool outside.")
```

 # Problem 2:

```
num = 3
if num >= 0:
print("The number is positive.")
else:
print("The number is negative”)
```

 # Problem 3 (nested if)

```
x = 5
if x > 2:
if x < 10:
print("x is between 2 and 10.")
```

 # Problem 4 (nested if)

```
x = 10
if x > 5:
print("x is greater than 5")
if x < 15:
print("x is also less than 15")
```

Read the code and answer the questions below:

```
x = 5
if x > 3:
print("x is greater than 3")
```

a) Identify the issue with the indentation in the code.
b) Describe the correct indentation for the code to execute properly.
c) Explain what will happen if the indentation is corrected.