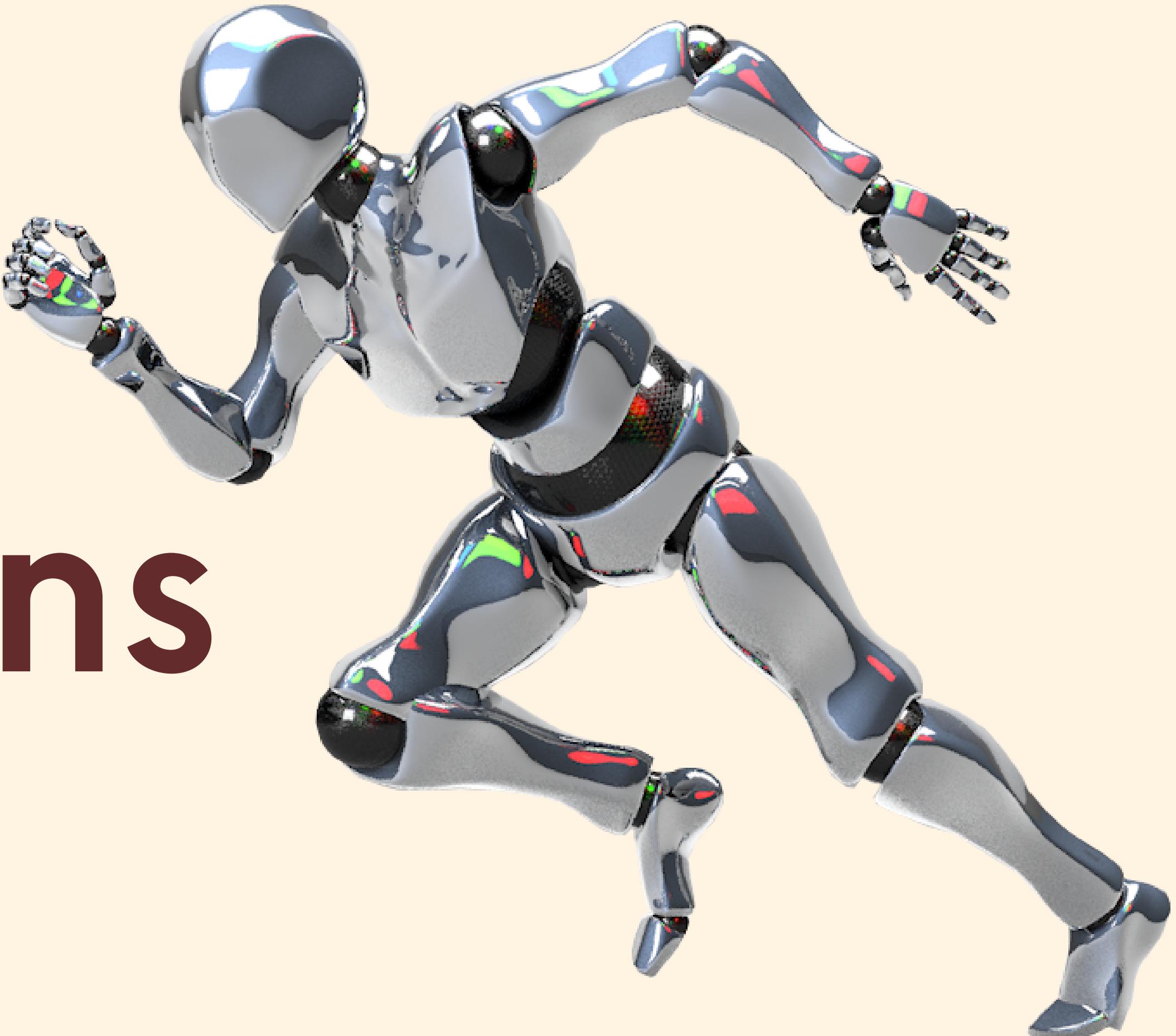
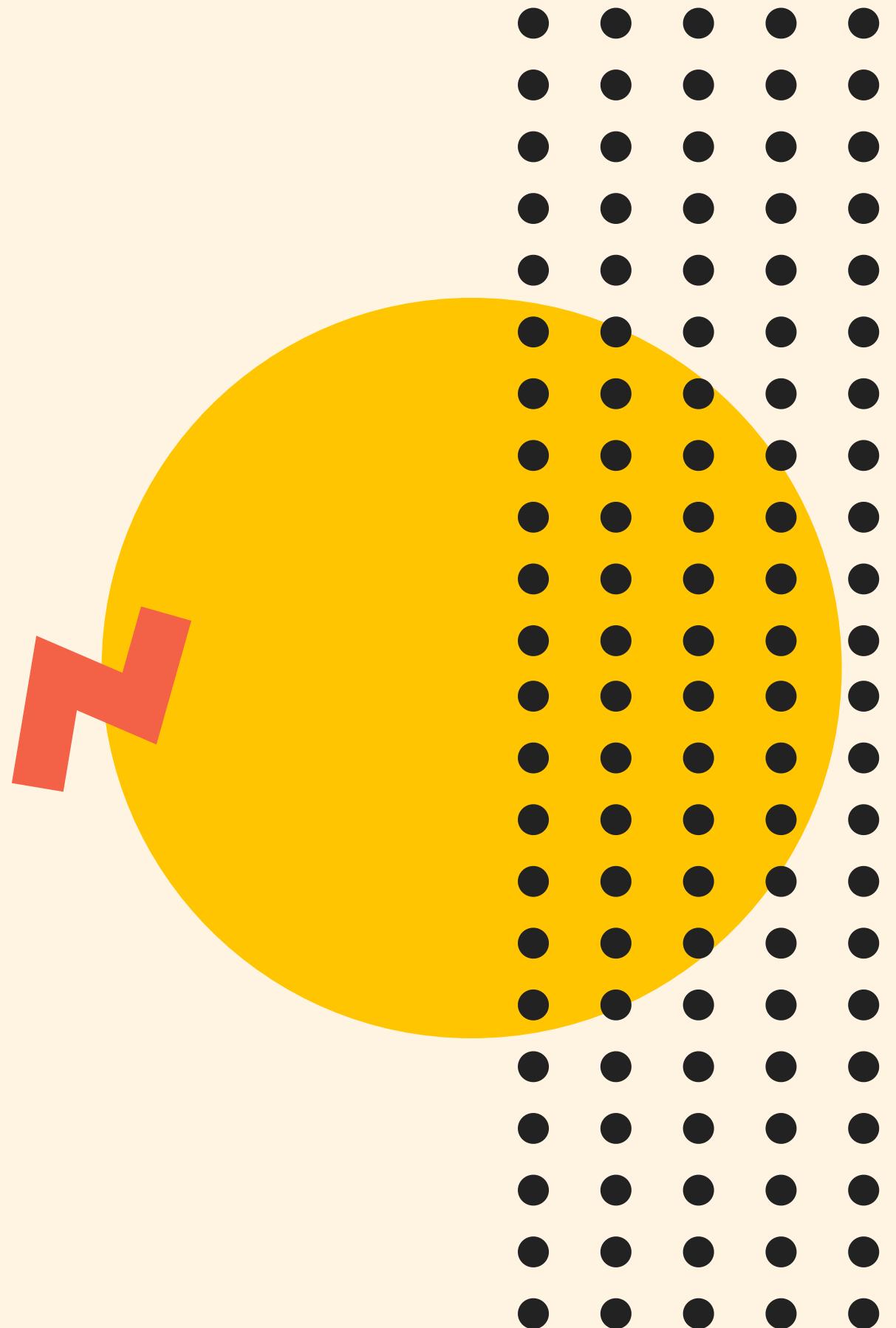
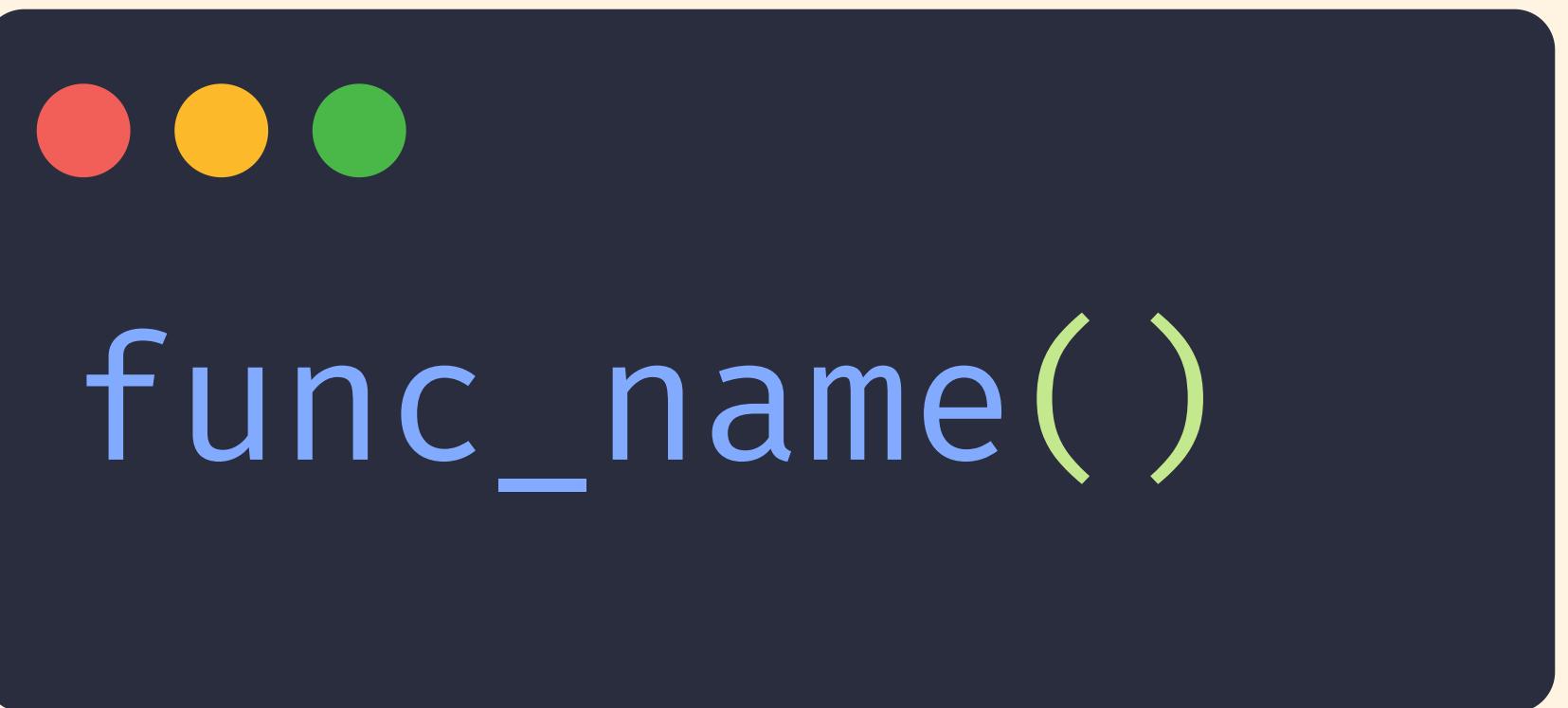


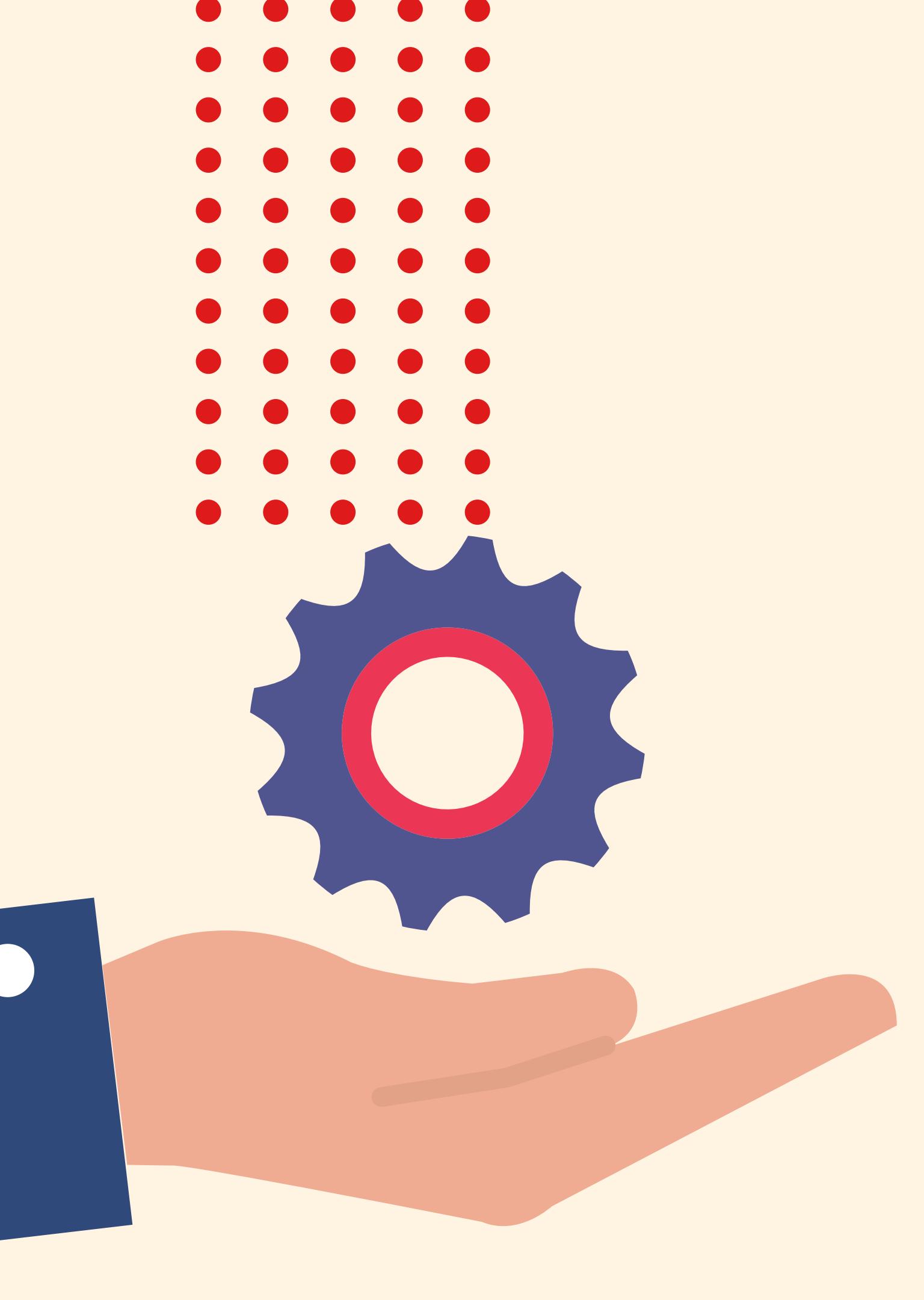
Functions



Functions

**FUNCTIONS ARE REUSABLE
ACTIONS THAT HAVE A NAME**





Functions

WHY USE THEM?

- We can use functions to prevent code duplication. Keep code DRY
- Functions help us abstract away code, breaking a complex program down into small pieces.

Define

**BEFORE WE CAN USE A
FUNCTION, WE MUST DEFINE
IT AND GIVE IT A NAME.**

Execute

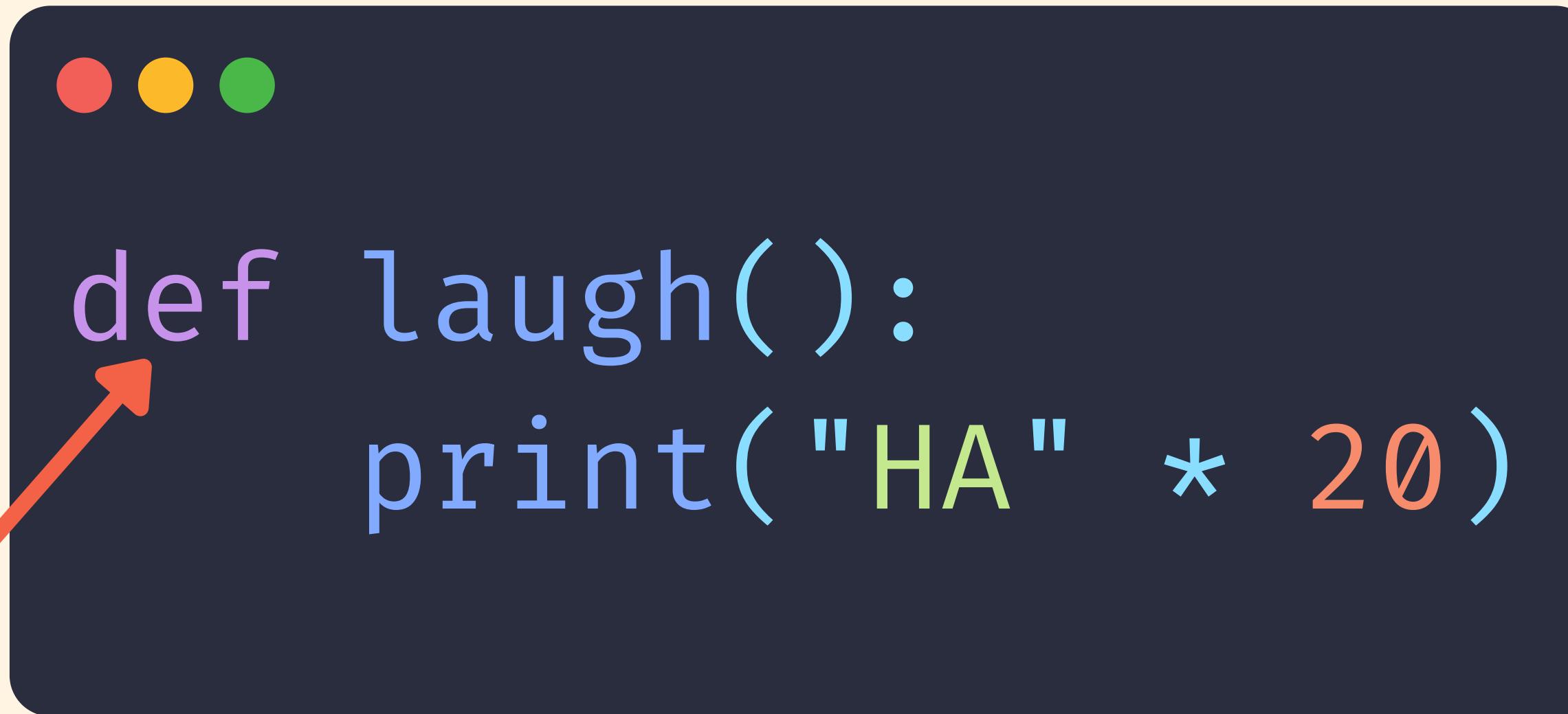
**ONCE PYTHON "KNOWS"
ABOUT OUR FUNCTION, WE
CAN CALL IT ANYTIME.**

Defining A Function



```
def laugh():
    print("HA" * 20)
```

Defining A Function



```
def laugh():
    print("HA" * 20)
```

def
keyword

Defining A Function

```
def laugh()  
    print("HA" * 20)
```

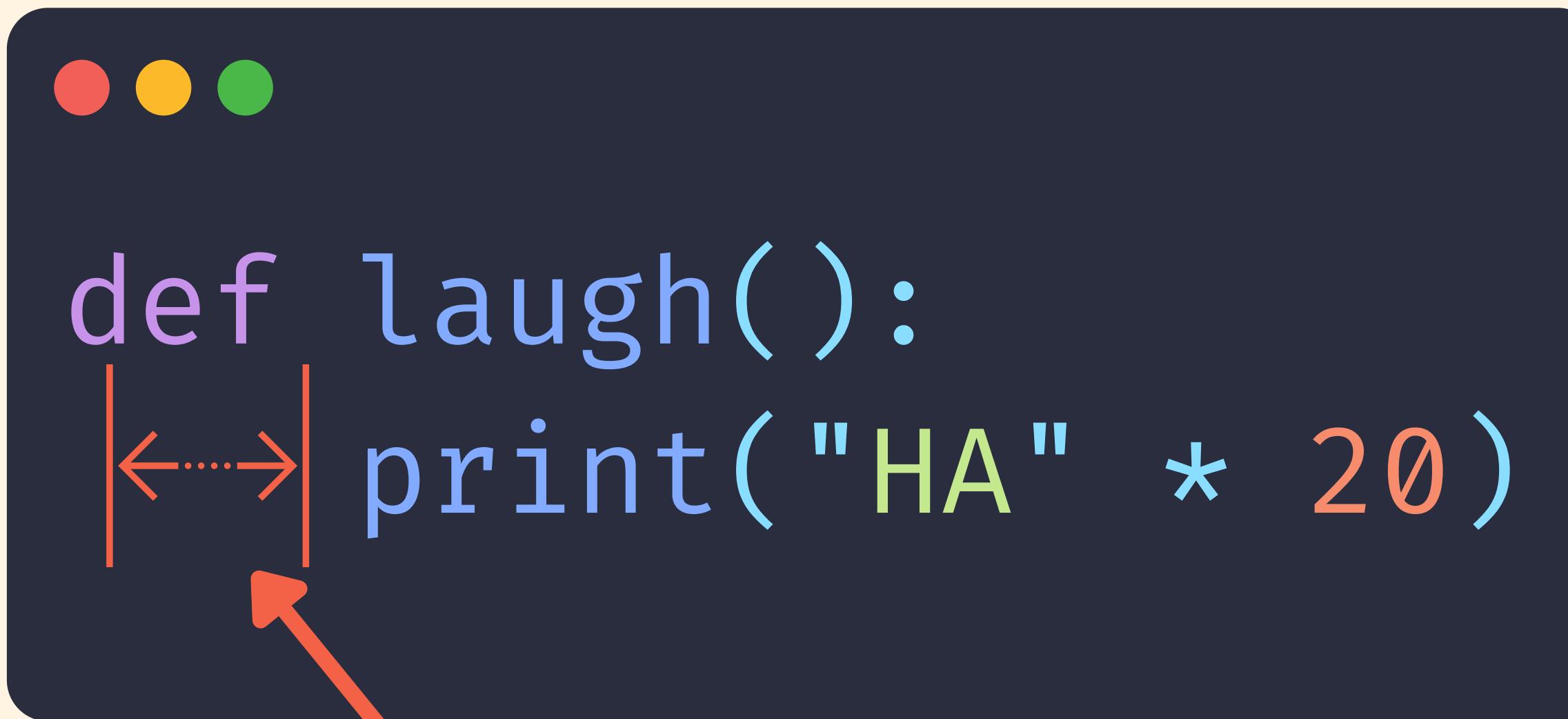
parens

Defining A Function

```
def laugh():  
    print("HA" * 20)
```

colon

Defining A Function



```
def laugh():
    print("HA" * 20)
```

indentation!

Calling A Function



laugh()

HAHAHAHAHAHAHAHAHAHA

HAHAHAHAHAHAHAHAHAHA



Function Name

Parentheses



Function Name

Argument

Parentheses

Arguments!



```
def laugh(intensity):  
    print("HA" * intensity)
```



laugh(2)

HAHA

laugh(10)

HAHAHAHAHAHAHAHAHA

Arguments!



```
def divide(x,y):  
    print(x/y)
```

Order of the arguments matters!



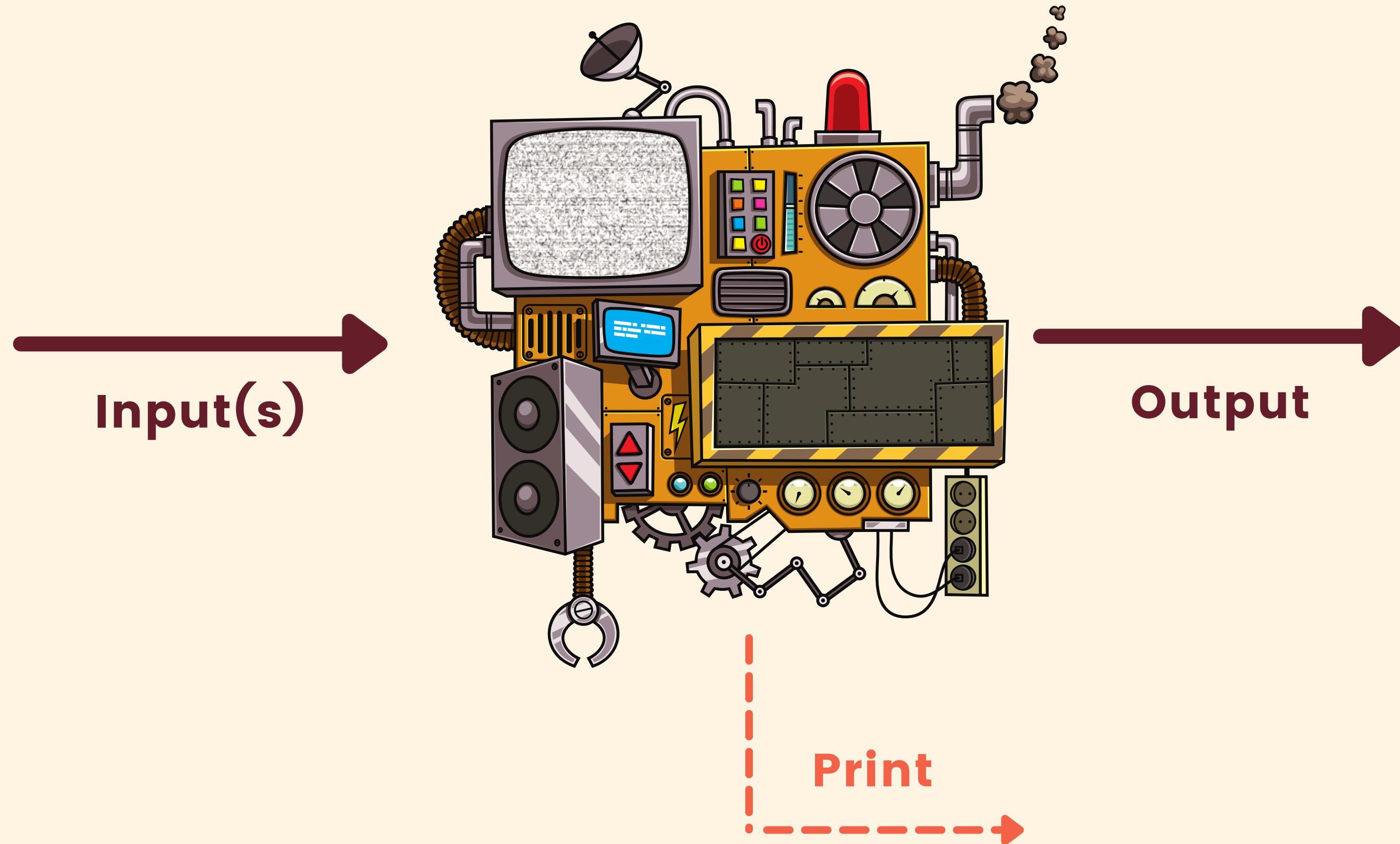
divide(12,3)

4.0

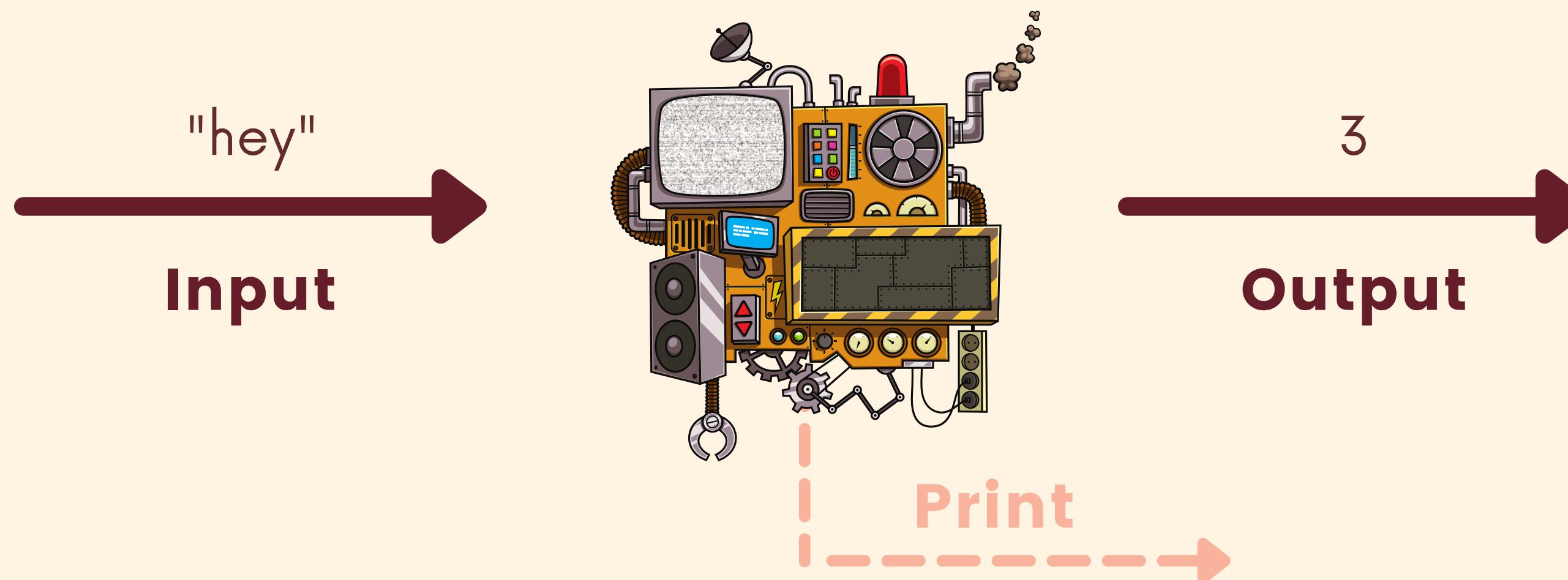


divide(3,12)

0.25



```
● ● ●  
> num = len("hey")  
> num  
3
```

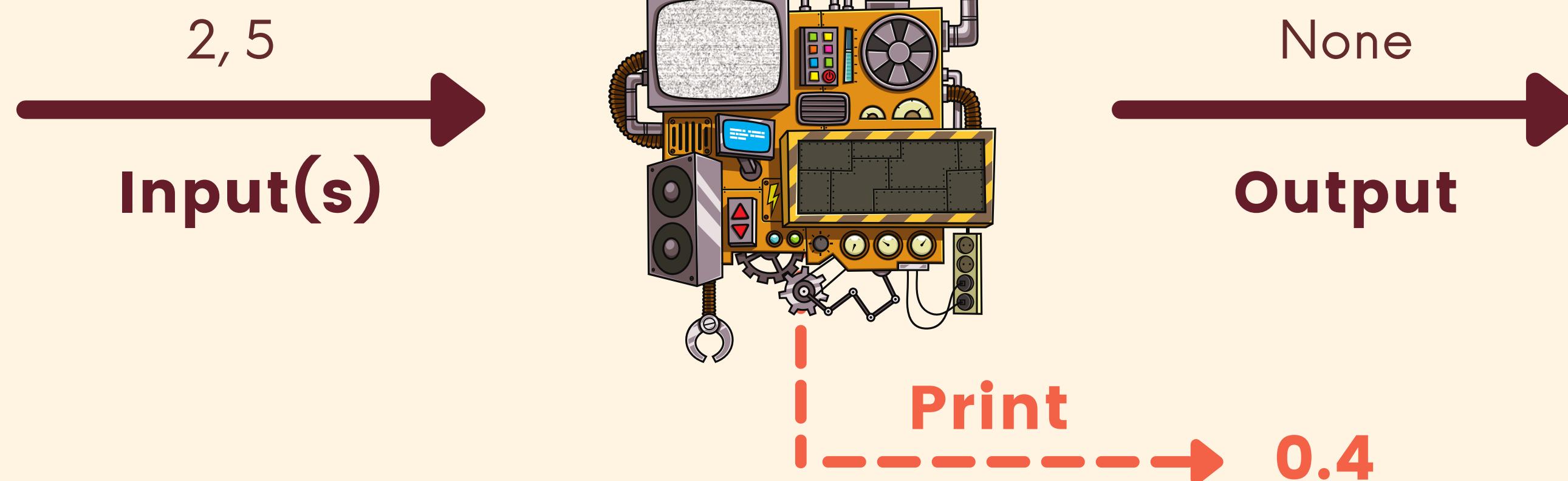


• • •

```
def divide(x,y):  
    print(x/y)
```

• • •

```
> n = divide(2,5)  
0.4  
> n  
>
```



The Return Keyword



```
def divide(x,y):  
    return x/y
```

Outputs whatever value comes after return keyword
Ends the execution of a function



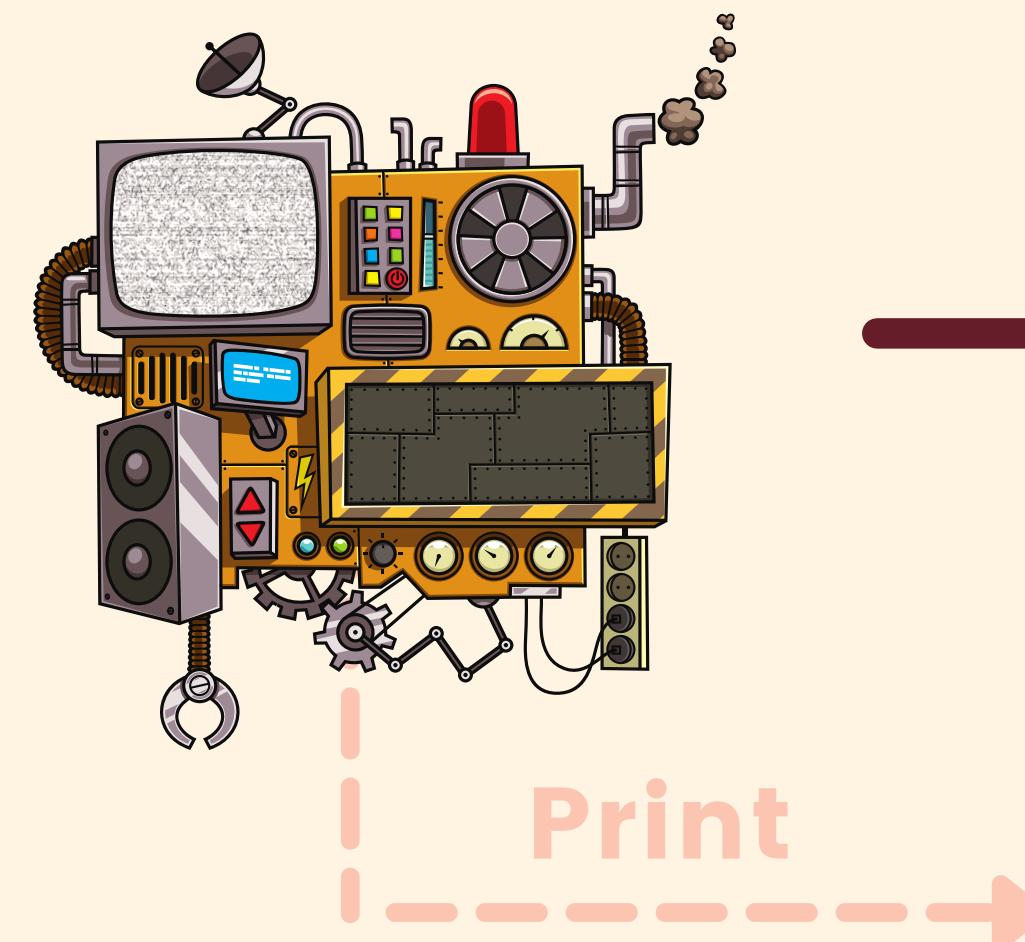
```
def divide(x,y):  
    return x/y
```



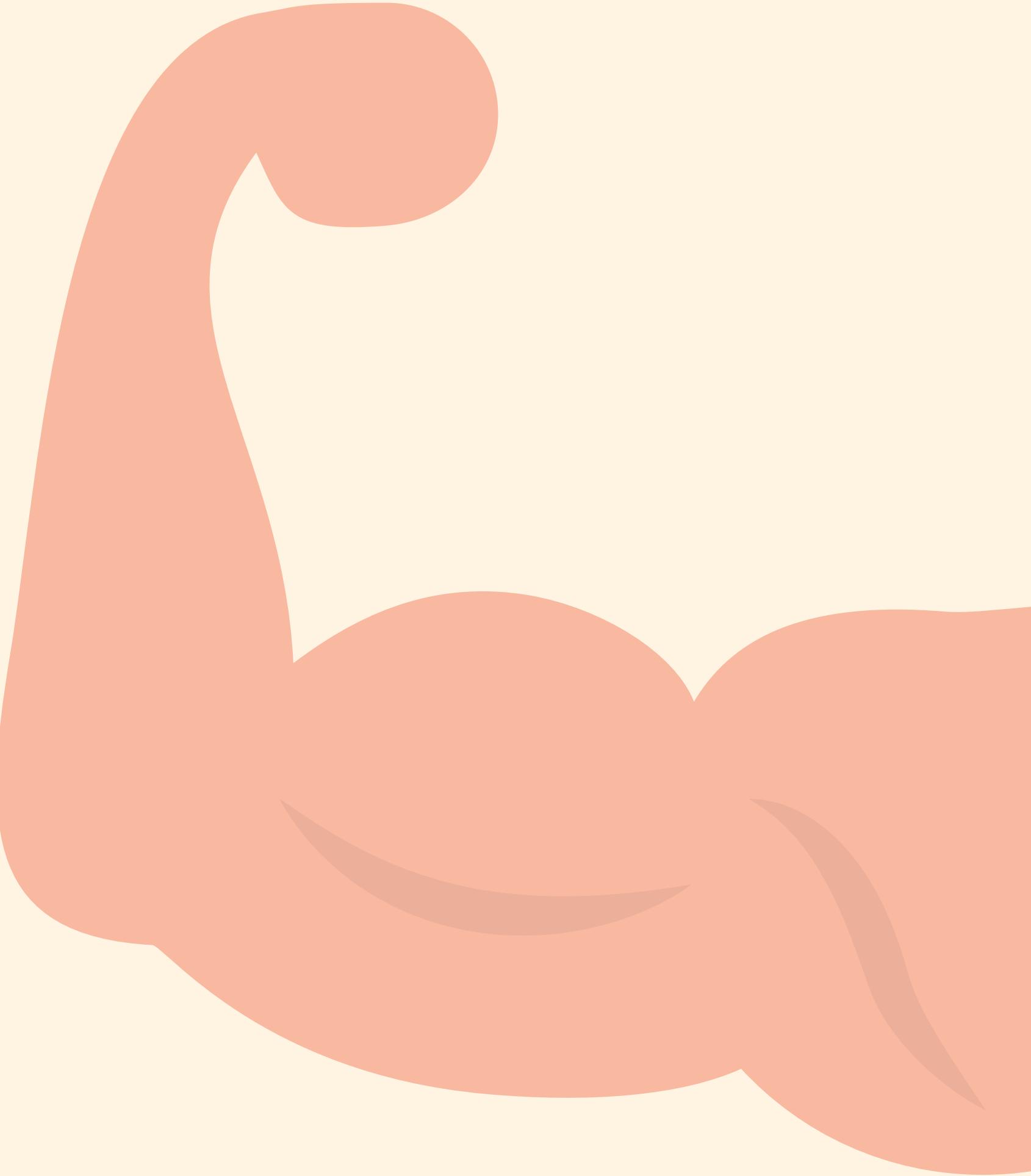
```
> n = divide(2,5)  
> n  
0.4
```

2, 5
Input(s)

0.4
Output



Practice



Default Parameters



```
def laugh(intensity=10):  
    print("HA" * intensity)
```

To give a parameter a default value if no argument is provided, simply add the default using this format:

parameter=value