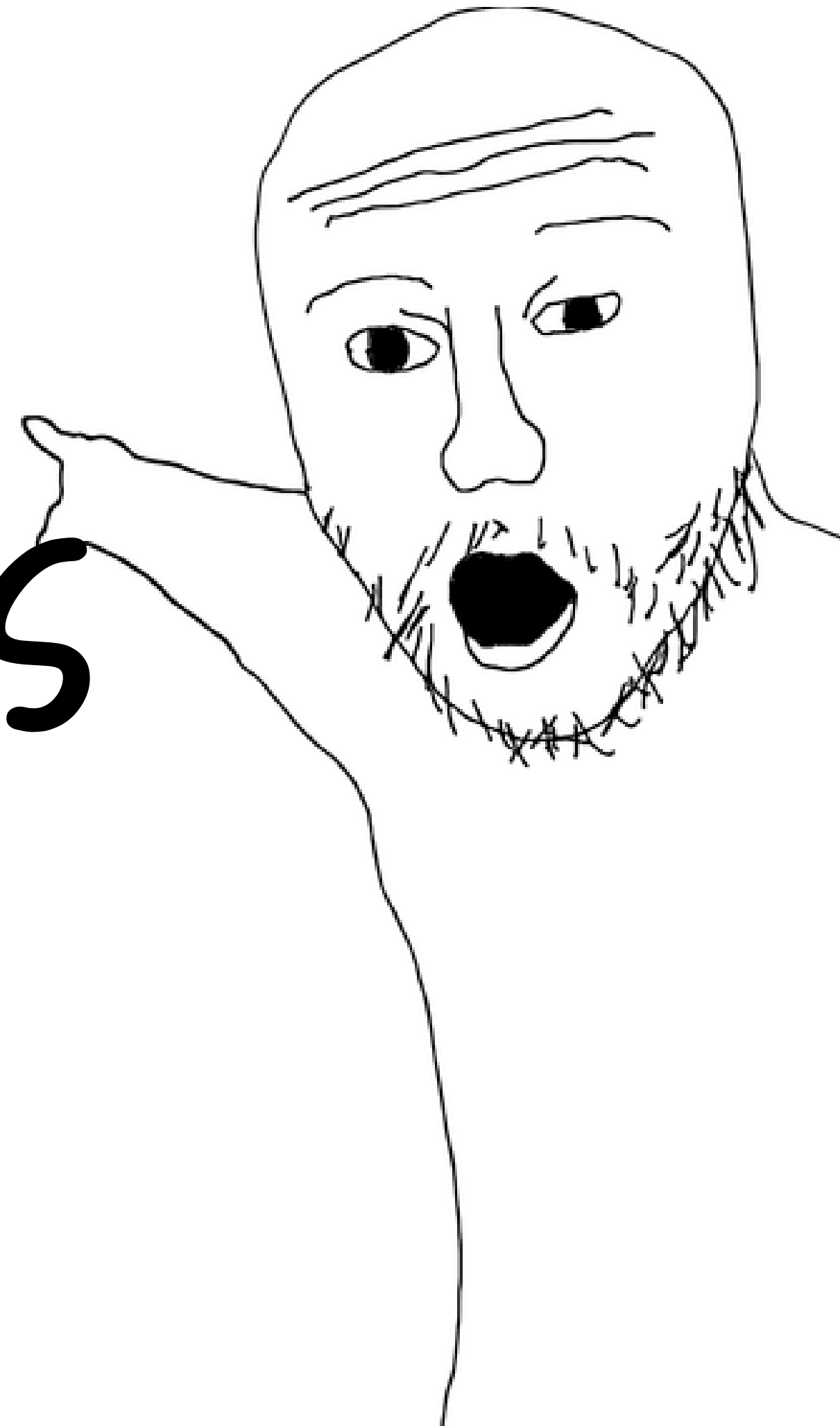


JAM CODERS

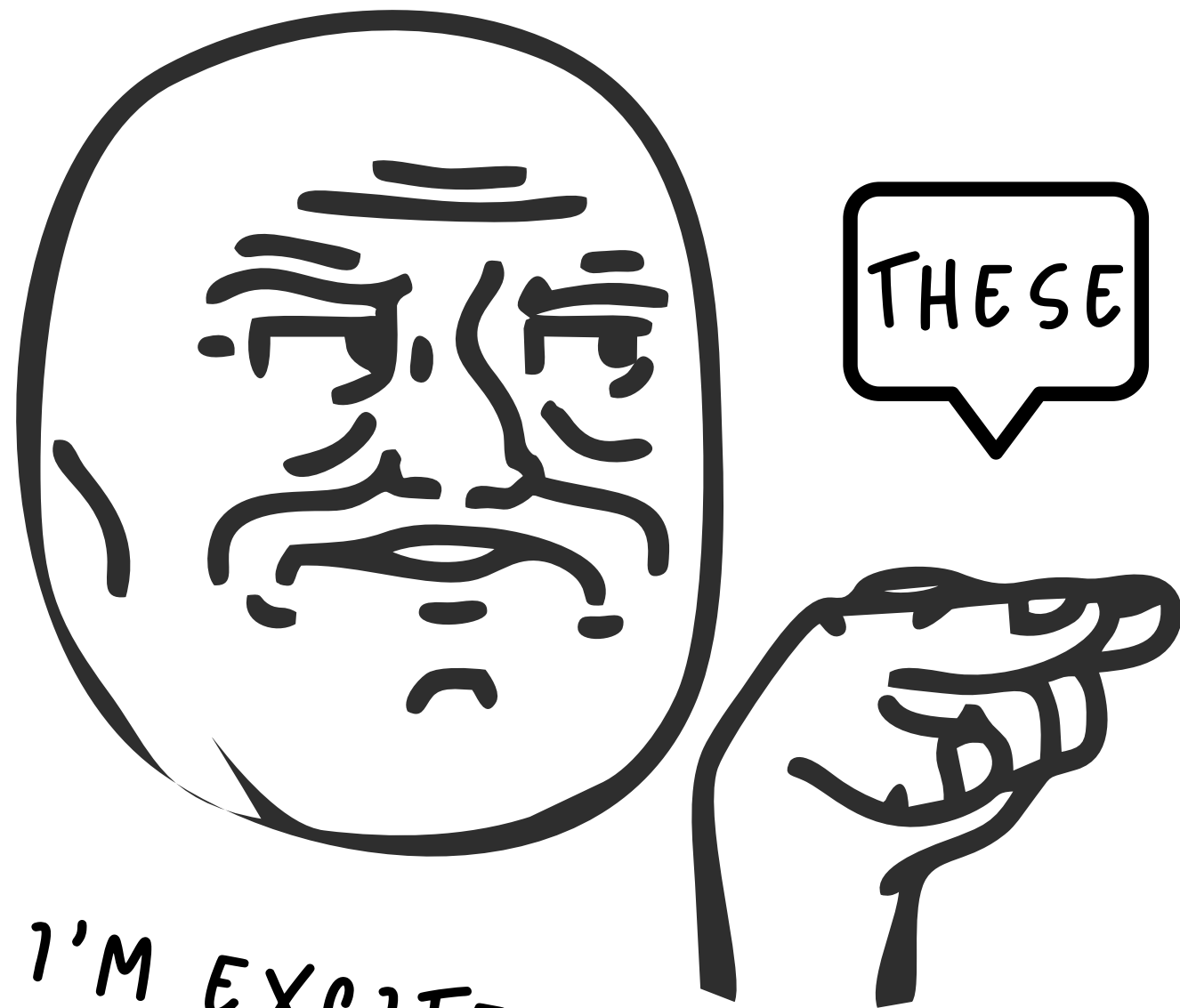


FUNCTIONS AGAIN??

YAAAAAYYYYYY!!



WHAT ARE WE COVERING?



I'M EXCITED...

LIST RECAP

.LOWER(),
.UPPER(),

TYPECASTING

.ISDIGIT()

INPUT()

PASS

WHAT DO YOU REMEMBER ABOUT LISTS IN PYTHON?

01

Defining a List:

```
my_list = []  
my_list = [1, 3, 4, 5, 6]  
my_list = ["hello", "123", True, 1]
```

Accessing a list:

1. **By Index:**

```
for i in range(len(my_list)):  
    print(my_list[i])
```

2. **By element**

```
for elem in my_list:  
    print(elem)
```

02



WHAT IS THE DIFFERENCE BETWEEN
THE TWO WAYS OF ACCESSING A
LIST ELEMENT?



03

Adding a value to our list:

```
my_list.append("hey")
```

WHAT IS TYPECASTING?

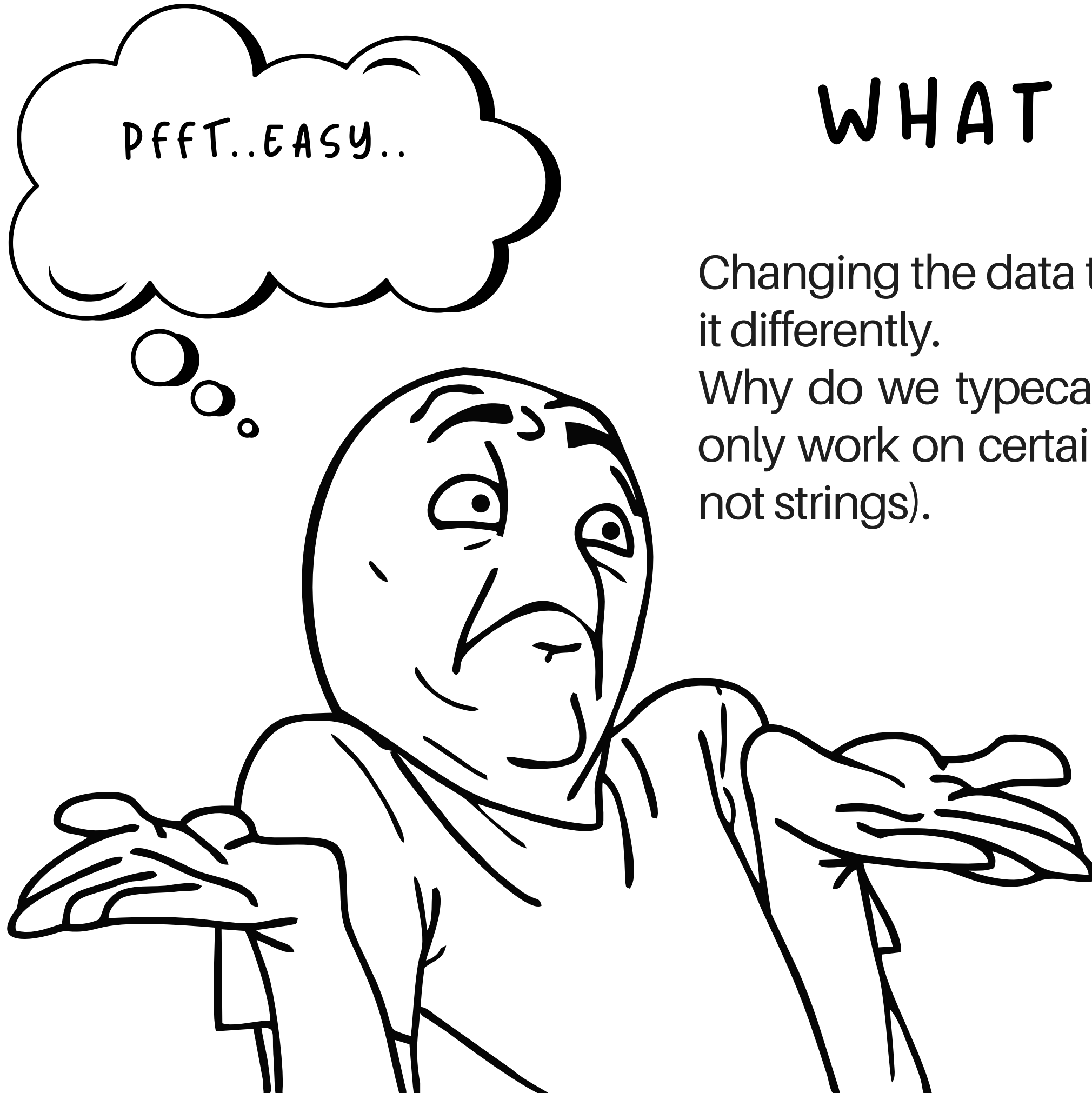
PFFT..EASY..

Changing the data type of a variable so we can use it differently.

Why do we typecast? To perform operations that only work on certain types (like math on numbers, not strings).

EXAMPLE

```
age = "12"  
print(type(age)) #str  
print(age + 20)  #Error  
age = int(age)  
print(type(age)) #int  
print(age + 20)  #32
```

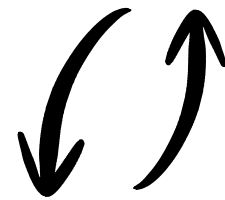


INPUT

WHAT ARE THEY FOR?

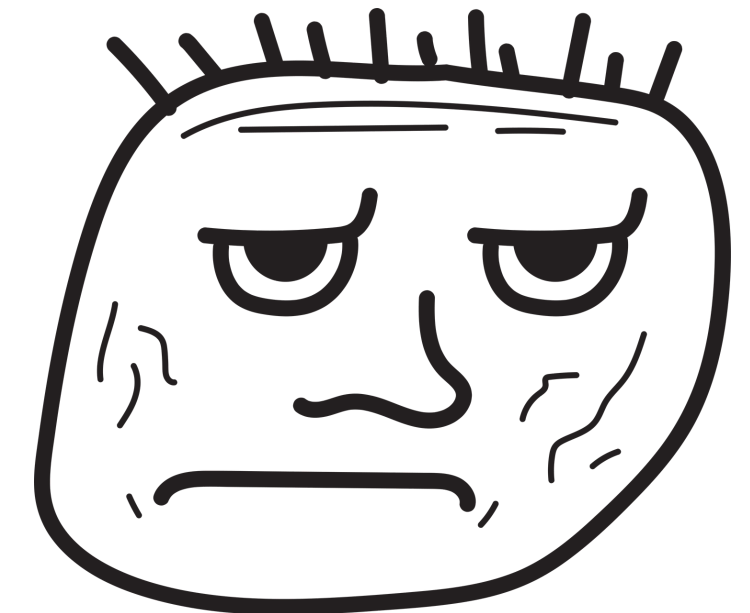
To accept information from the user - not just the programmer.

⚠ Remember: `input()` always returns a string, even if the user types a number.



FOR EXAMPLE

```
name = input("What is your name")  
print("hi " + name)
```



.lower(), .upper()

STRING.lower()

changes string to lower case.

```
my_string = "Anushka"  
my_string = my_string.lower()  
print(my_string) #anushka
```

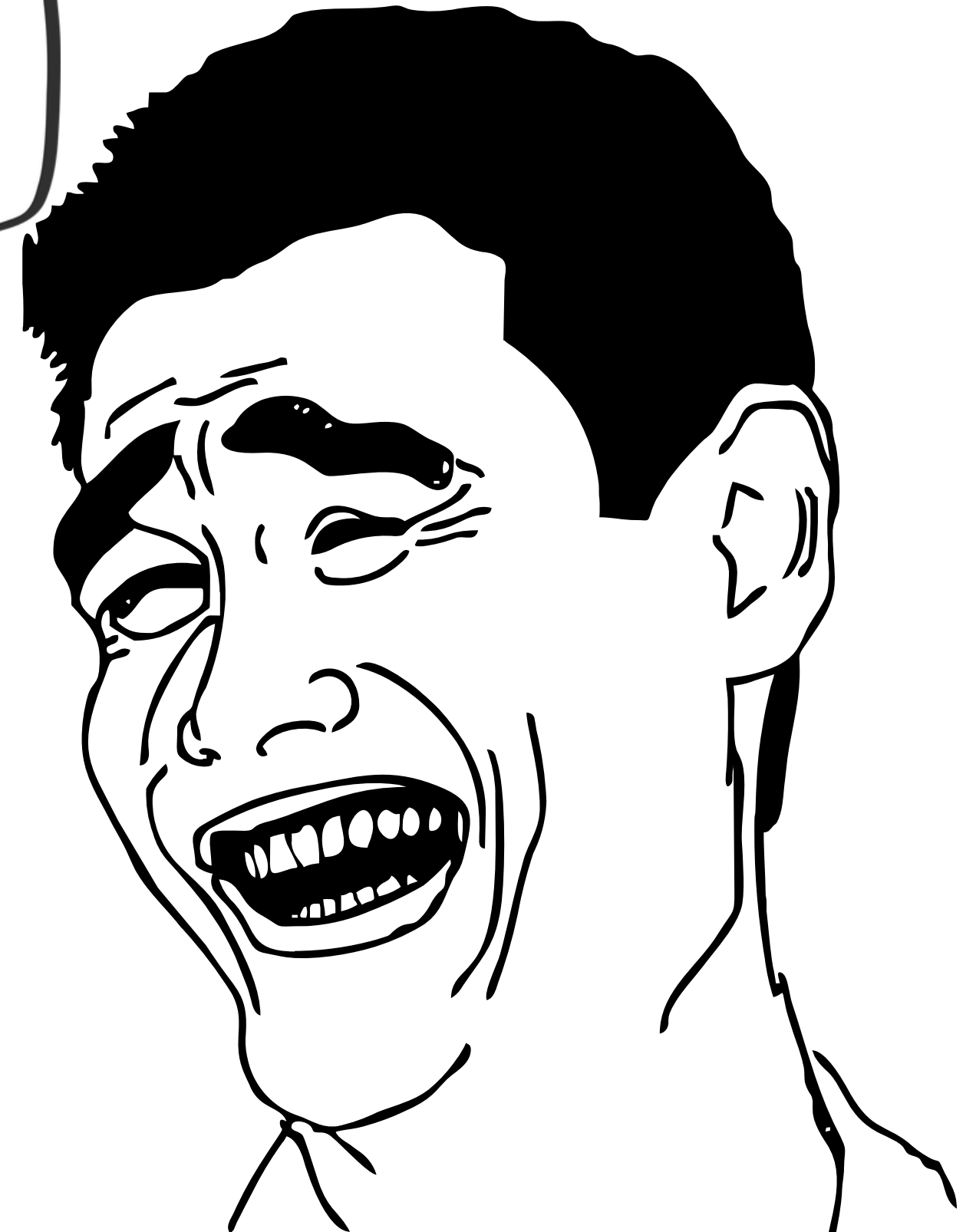
STRING.upper()

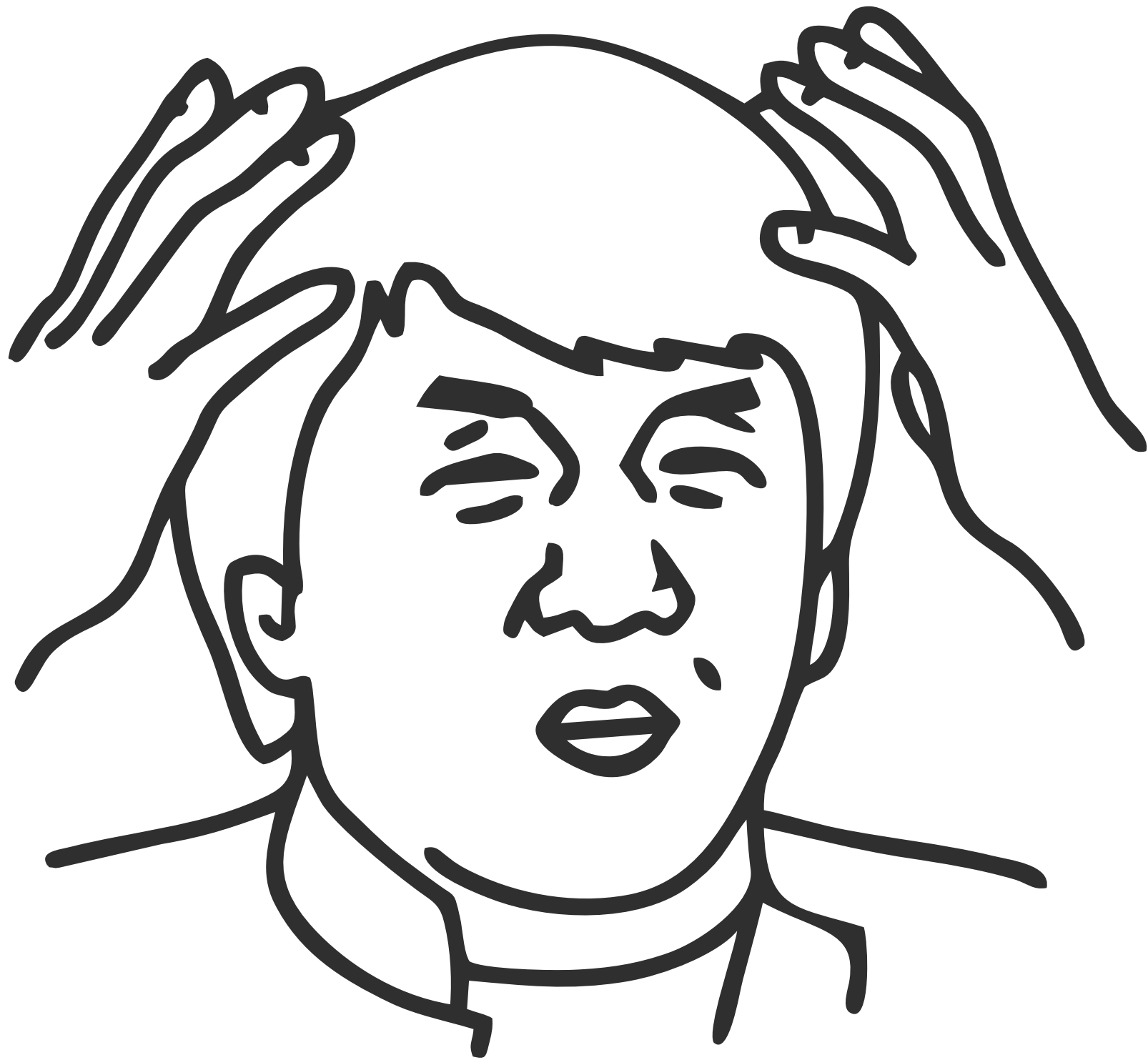
changes string to upper case.

```
my_string = "Sarika"  
my_string = my_string.upper()  
print(my_string) #SARIKA
```

I GET IT
NOW!!

SO EASY!





Checks whether a string contains only digits. **Returns True or False.**

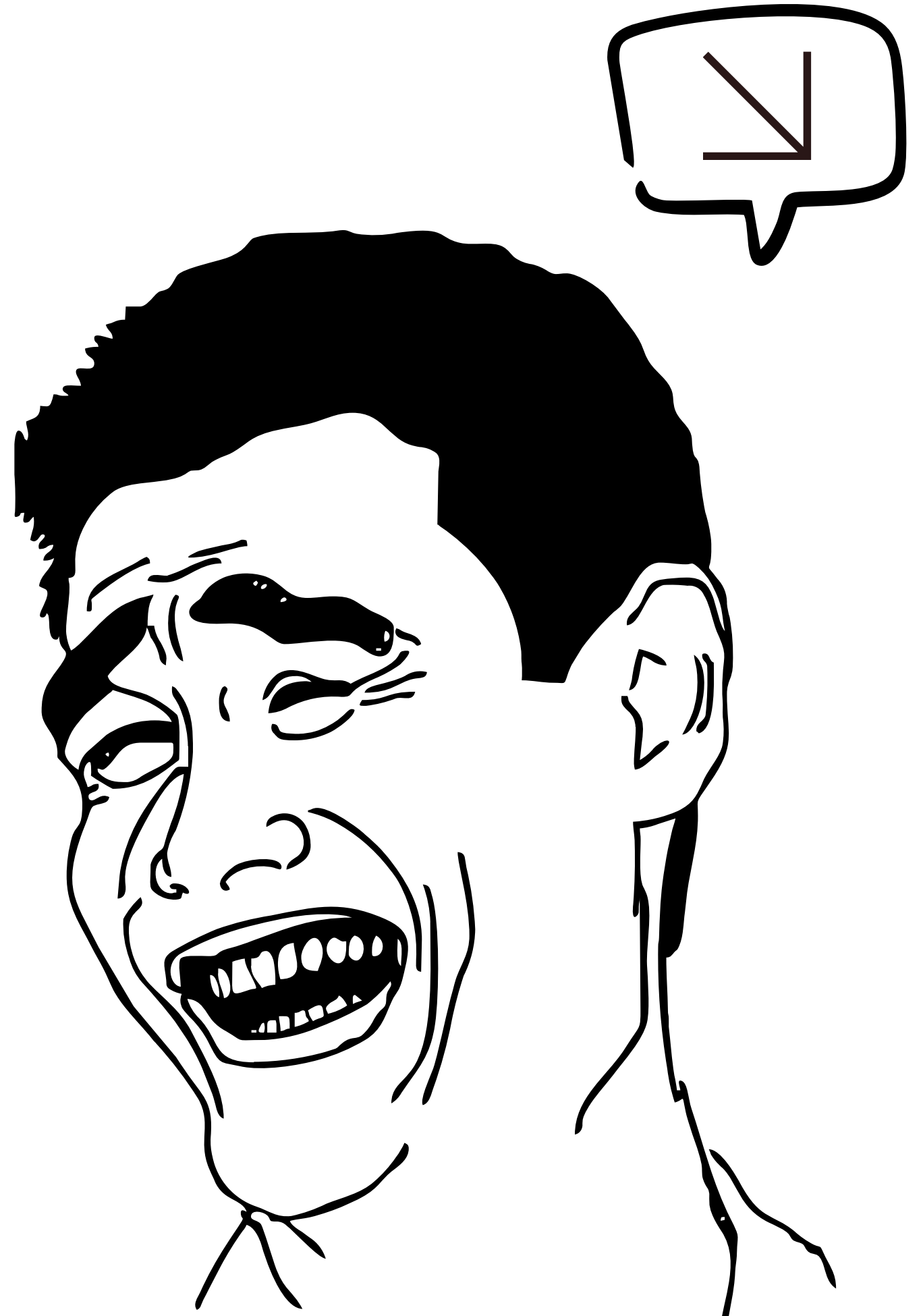
```
print("011".isdigit()) #True  
print("hey".isdigit()) #False  
print("778h".isdigit()) #False  
print("4%3".isdigit()) #False  
print("-3".isdigit()) #False
```

PASS

A **placeholder** used inside a function (or loop/if-block) when you haven't written the code yet, it tells Python "do nothing here, but don't error out."

```
def my_function(x,y):  
    pass
```

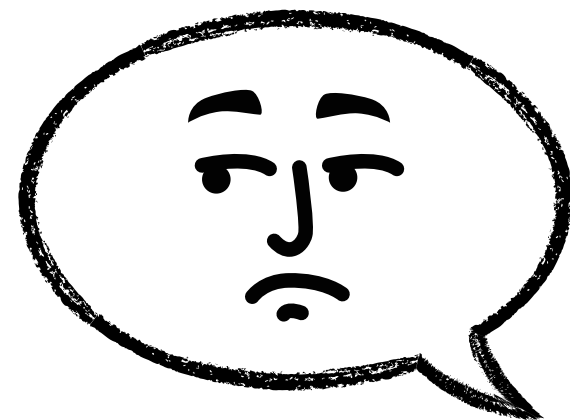
THAT IS ALL!!!



WELL DONE

THANK
YOU!

ANY QUESTION?



JUST
SAY NO!

