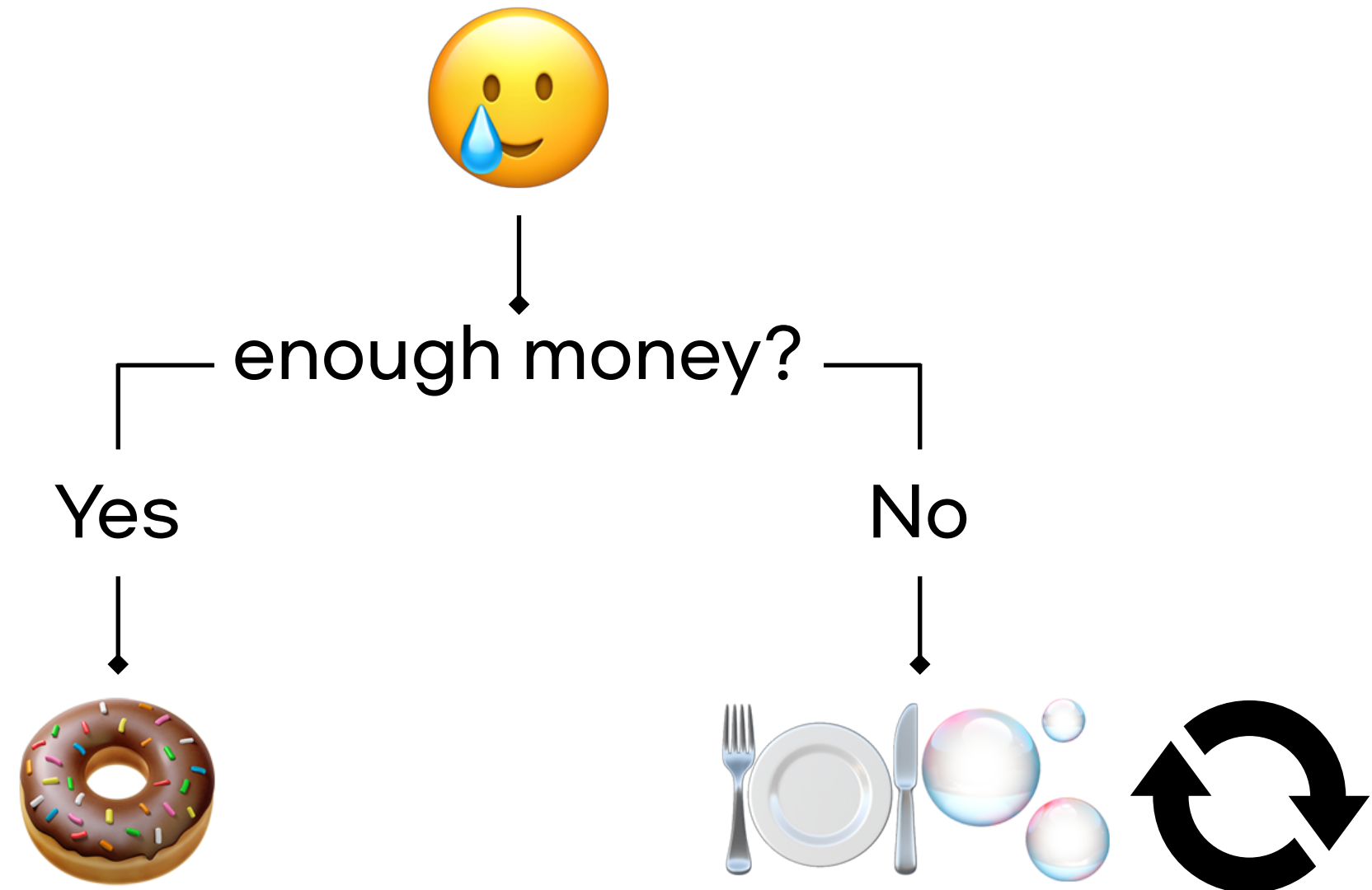


Control Flow & Function

Control Flow



Control Flow

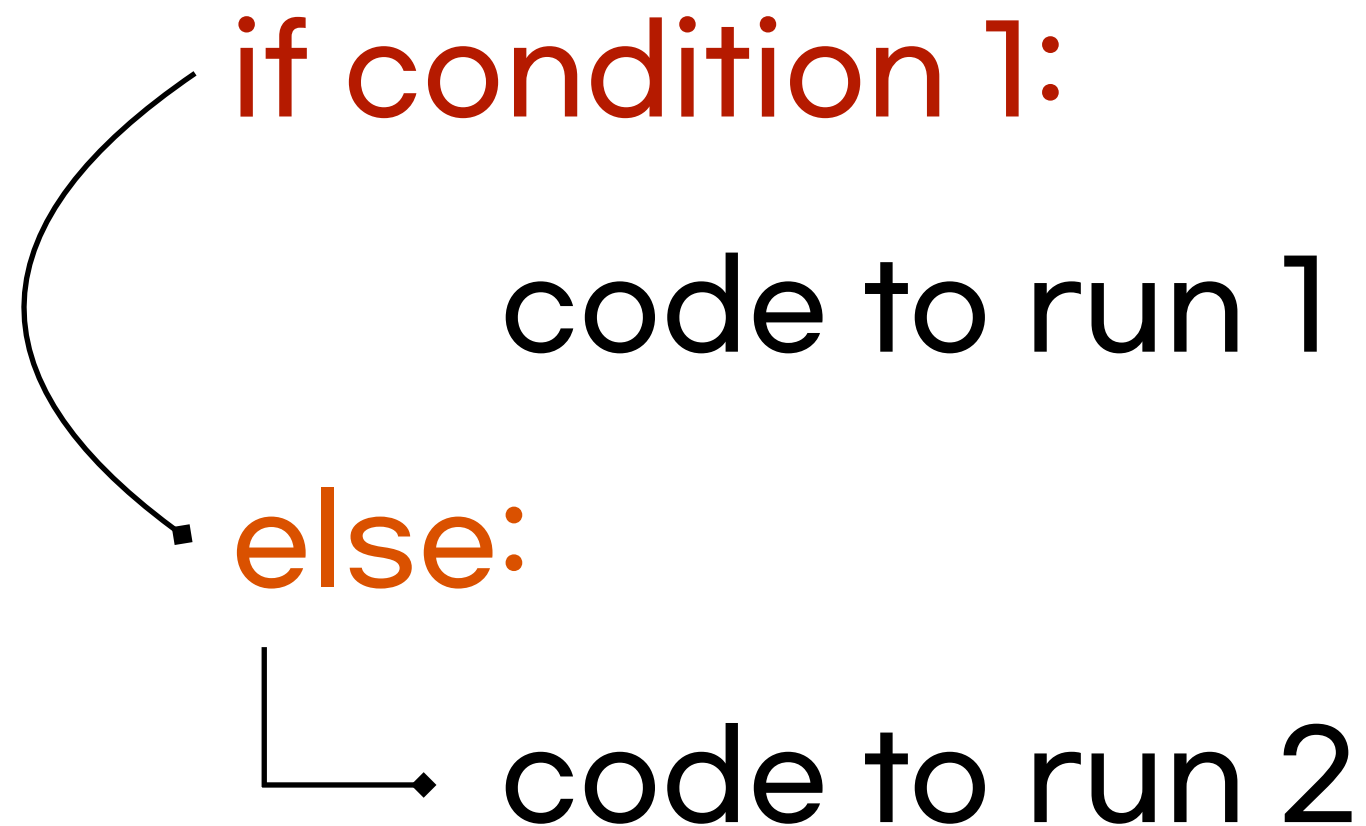


If Statement

if condition:

└ code to run

If Statement



If Statement

if condition 1:

code to run 1

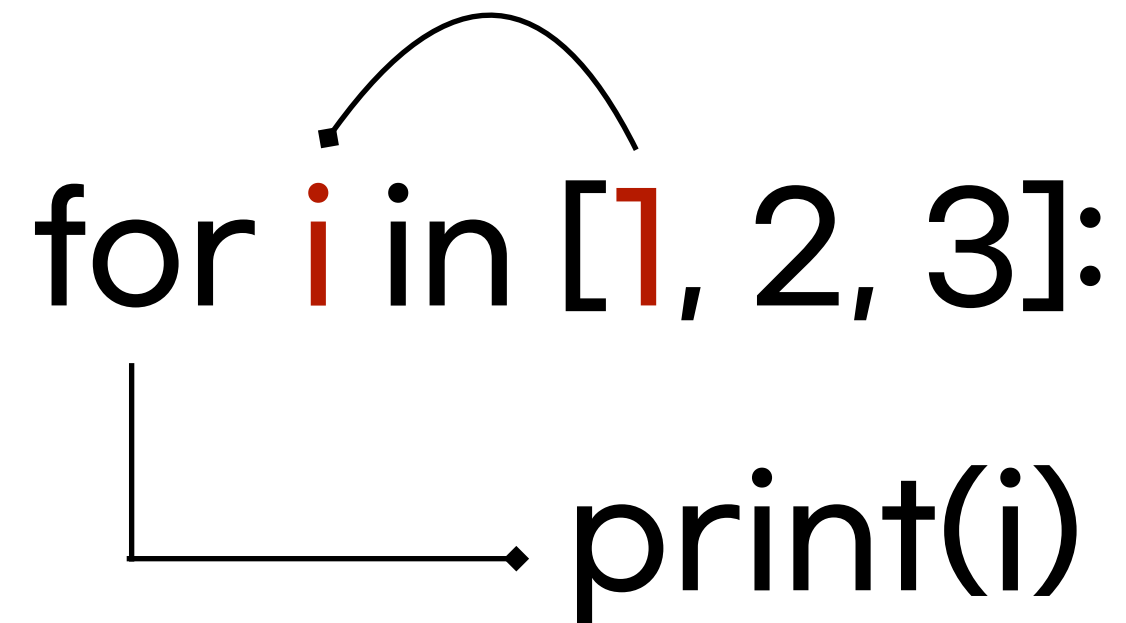
elif condition 2:

code to run 2

else:

└─ code to run 3

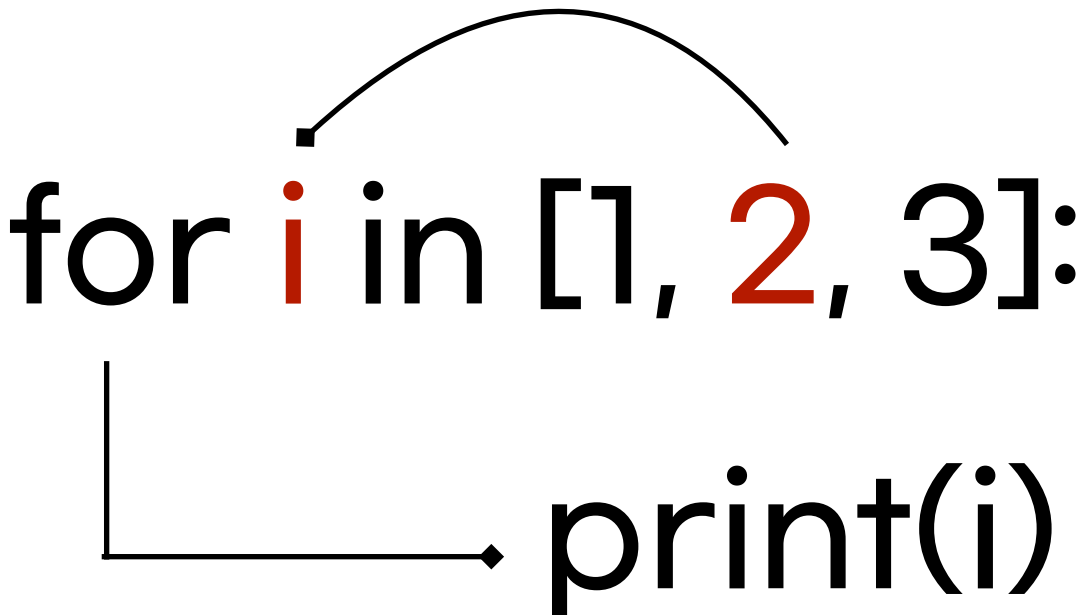
For Loop



```
for i in [1, 2, 3]:  
    print(i)
```


The diagram illustrates a for loop. The variable `i` is highlighted in red. A curved arrow points from the closing bracket of the list `[1, 2, 3]` back to the variable `i`, indicating the loop's iteration. A right-angle arrow points from the colon at the end of the first line to the `print(i)` statement on the second line, showing the code block associated with the loop.

For Loop



```
for i in [1, 2, 3]:  
    print(i)
```


For Loop



```
for i in [1, 2, 3]:  
    print(i)
```

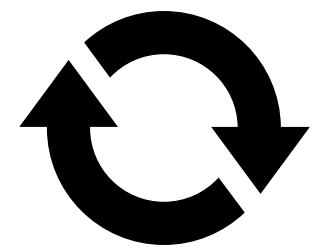
The diagram illustrates the execution of a for loop. A curved arrow originates from the right side of the list `[1, 2, 3]` and points to the variable `i`, representing the assignment of each element in the list to the loop variable. A second arrow originates from the `i` in the loop header, moves vertically down, and then horizontally right to point at the `i` argument of the `print(i)` function, indicating that the current value of `i` is passed to the function for printing.

While Loop

while condition:



code to run



List Comprehension

a = [0, 1, 2, 3, 4]



b = [1, 2, 3, 4, 5]

List Comprehension

```
b = []
```

```
for i in a:
```

```
    b.append(i + 1)
```

List Comprehension

```
b = [for i in a]
```



```
for i in a:
```

```
    b.append(i + 1)
```

List Comprehension

The diagram illustrates the transformation of a loop into a list comprehension. It features three lines of code. The top line is a list comprehension: `b = [i + 1 for i in a]`, where `i`, `+`, and `1` are highlighted in red. The middle line is a loop header: `for i in a:`, rendered in gray. The bottom line is a loop body: `b.append(i + 1)`, where `i` and `1` are rendered in gray. A vertical arrow points from the `i` in the loop header to the `i` in the list comprehension. A horizontal line connects the end of the loop body to the start of the list comprehension, with a downward arrow pointing to the `+` and `1` in the list comprehension.

```
b = [i + 1 for i in a]
```

```
for i in a:
```

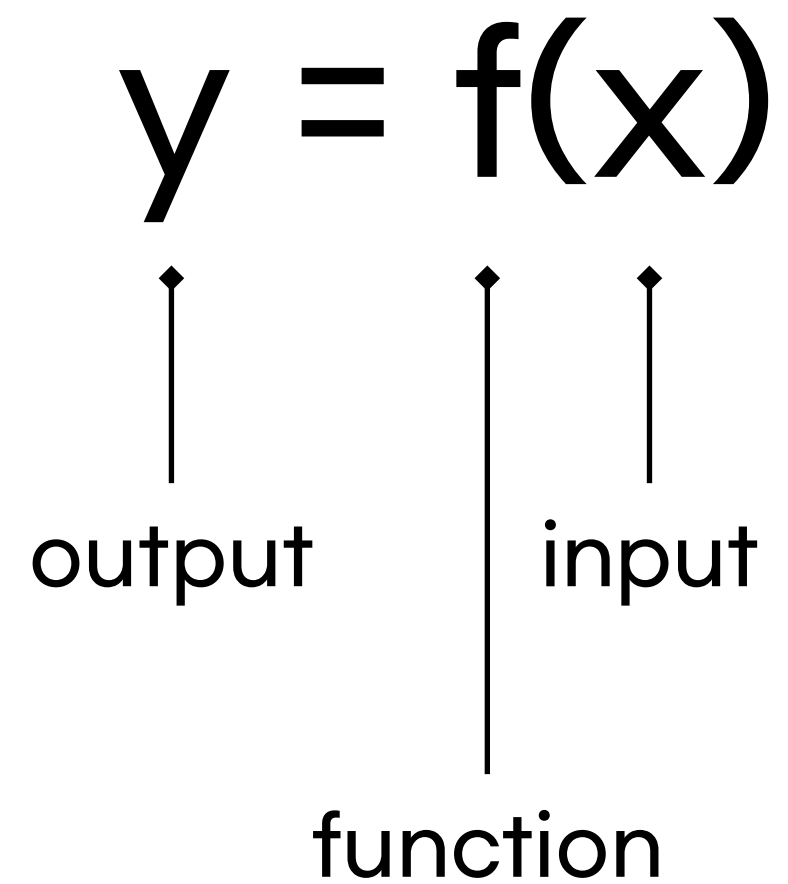
```
    b.append(i + 1)
```

List Comprehension

`a = [0, 1, 2, 3, 4]`

`b = [i + 1 for i in a]`

Function



Function

```
def function_name(input):  
    code_to_run  
    return output
```

Local Scope

```
def function_name(input):  
    code_to_run  
    return output
```

Global Scope

```
a = "global variable"
```

```
def function_name(input):  
    code_to_run  
    return output
```