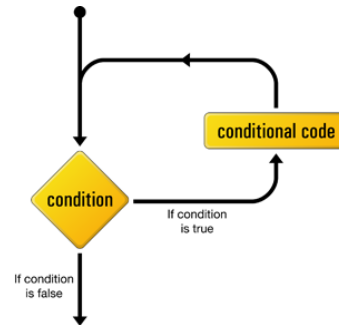


Model 1 while Statements

A **while** statement is a general loop that continues to run while the condition is true. Run the following code, and record the output.

```
i = 0
while i < 3:
    print("the number is", i)
    i = i + 1
print("goodbye")
```



Questions (15 min)

Start time:

1. What must the value of the Boolean expression (after the **while**) be in order for the first print statement to execute?

2. Circle the statement that changes the variable *i* in the above code.

3. What happens to the value of the variable *i* during the execution of the loop?

4. Explain why the loop body does not execute again after “the number is 2” is output.

5. Reverse the order of the statements in the loop body:

```
while i < 3:
    i = i + 1
    print("the number is", i)
```

a) How does the order impact the output displayed by the **print** function?

b) Does the order impact the total number of lines that are output?

6. Identify three different ways to modify the code so that the loop only executes twice.

7. Describe the three parts of a `while` loop that control the number of times the loop executes.

8. Comment out the statement `i = i + 1`, and run the code. Then press Ctrl-C (hold down the Ctrl key and press C). Describe the behavior you see, and explain why it happened.

9. Consider the following program:

```
1 total = 0
2 count = 0
3
4 n = None
5 while n != 0:
6     n = float(input("Next number: "))
7     total += n
8     count += 1
9
10 print()
11 print("Total:", total)
12 print("Count:", count)
```

a) Which line initializes the loop variable?

b) Which line updates the loop variable?

c) Can you predict how many times the loop will run? Explain why or why not.

d) Does the code have the potential to loop forever? Explain why or why not.