Conditional Execution

To write useful programs, we almost need the ability to check the conditions and change the behaviour accordingly. Conditional statements give us this ability.

The simplest form is the <u>if</u> statement :-

??

```
if x > 0:
    print("x is positive")
```

Condition

Reserved word

```
if x > 0 :
    print('x is positive')
```

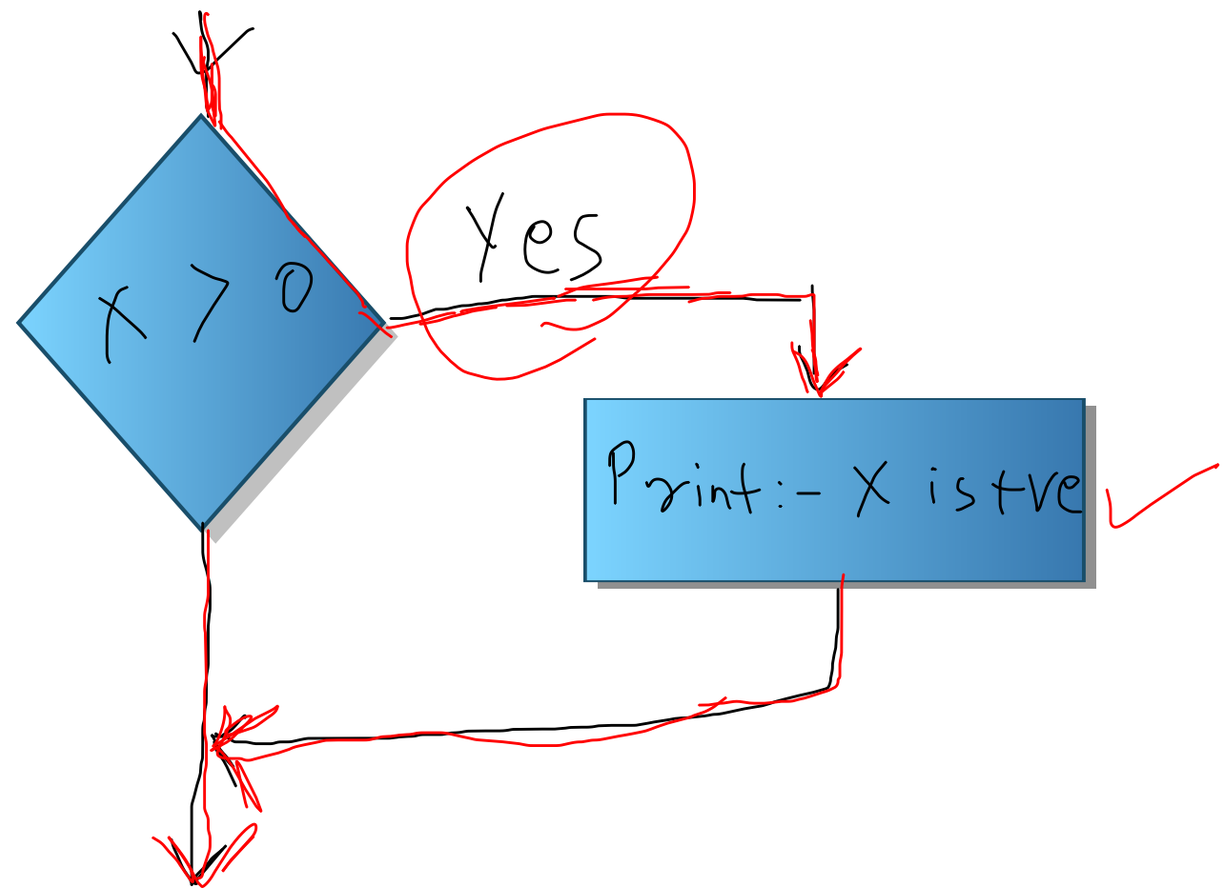
- The boolean expression after if-statement is called <u>condition</u>.

T/F

```
x = -1 ;    x > 0 → False
x = 2 ;     x > 0 → True
```

- We end the if statement with a colon character (:)

- The line(s) after the if-statement are indented. (4 SPACES)



If-logic

- If logical condition -> True, indented statement gets executed.
- If logical condition -> false, indented statement is skipped.
- Statements like this are called compound statements because they stretch across more than one line
- Pass statement, does nothing!

Alternative Execution

→ There are two possibilities.

→ Condition determines which ones gets executed.

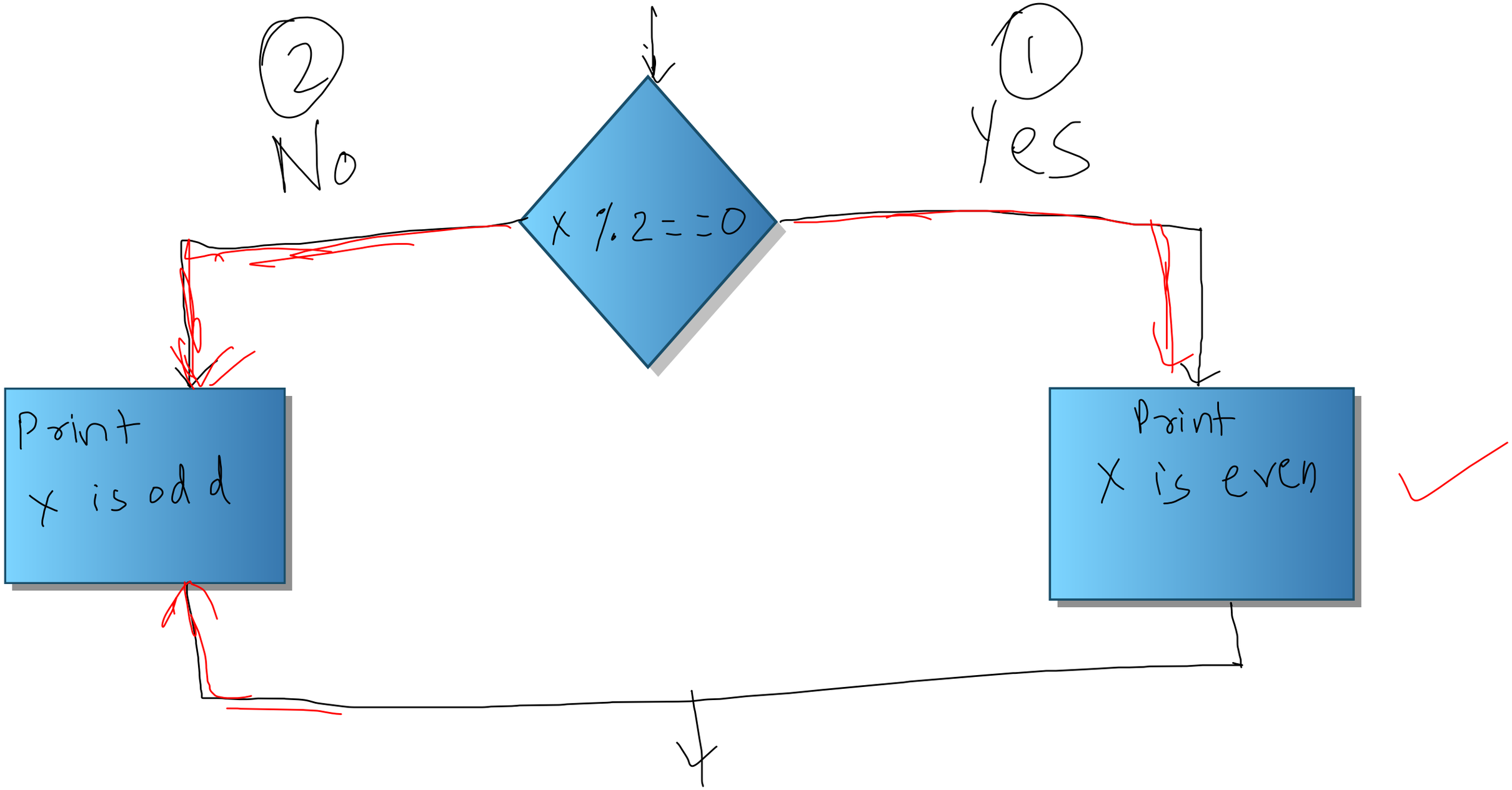
Remainder

```
if x % 2 == 0:
    print('x is even')
else:
    print('x is odd')
```

T

F

→ same



If - Then - Else Logic

- Since, the condition must either be true or false, exactly one of the alternatives will be executed.

- The alternatives are called branches, because they are branches in the flow of execution.