Boolean Expressions

Literals

1. Two possible values, True and False
2. Boolean Literals are **True** and **False**

Relational Operators

```
age = 24
pet = 'dog'
```

<	19 < age → True	pet < 'zebra' → True
<=	19 <= age → True	pet <= 'petra' → True
>	19 > age → False	pet > 'cat' → True
>=	19 >= age → False	pet >= 'monkey' → False
==	19 == age → False	pet == 'Dog' → False
!=	19 != age → True	pet != 'horse' → True

Be careful comparing float values for equality

```
x = .1 + .1 + .1
y = .3
x == y     →   False
```

Do this instead

```
import math
math.isclose(x, y)    →     True
```

Difference between *assignment* and *equality*

```
value = 25            # Assignment
value == 5 ** 3       # Equality
```

Checking for a range

```
0 <= value <= 100
```

Logical Operators

and **or** **not**

and

`P and Q` → Only True when P and Q are both True

X	Y	X and Y
T	T	T
T	F	F
F	T	F
F	F	F

Try with different vales for answer

```
answer < 10 and answer != 7
```

or

`P or Q` → Only False when P and Q are both False

X	Y	X or Y
T	T	T
T	F	T
F	T	T
F	F	F

Try with different vales for answer

```
answer < 10 and answer == 15
```

not

Only one operand
Changes True to False / False to True

X	not X
T	F
F	T

Try with different vales for answer

```
not answer < 17
```

Boolean Variables

```
score = 87
passed = score >= 70        →      True
not passed                  →      False
```

Control Structures

1. Sequence
2. Selection
3. Repletion

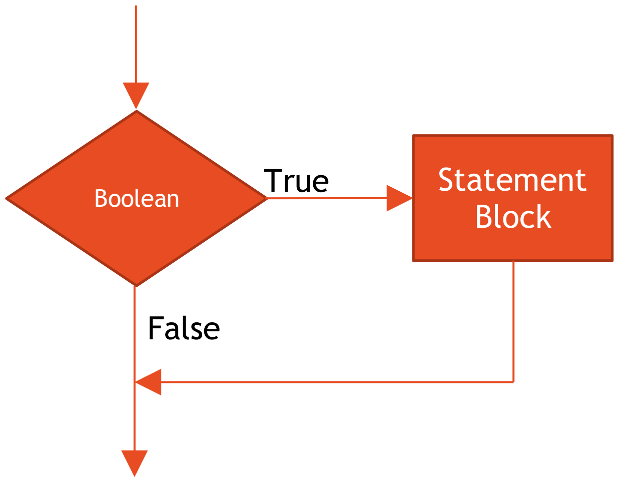
Sequence

We've gotten this for free

Selection – That's what we're going to do now

Select from multiple possible execution paths – Based on a Boolean condition

if statement



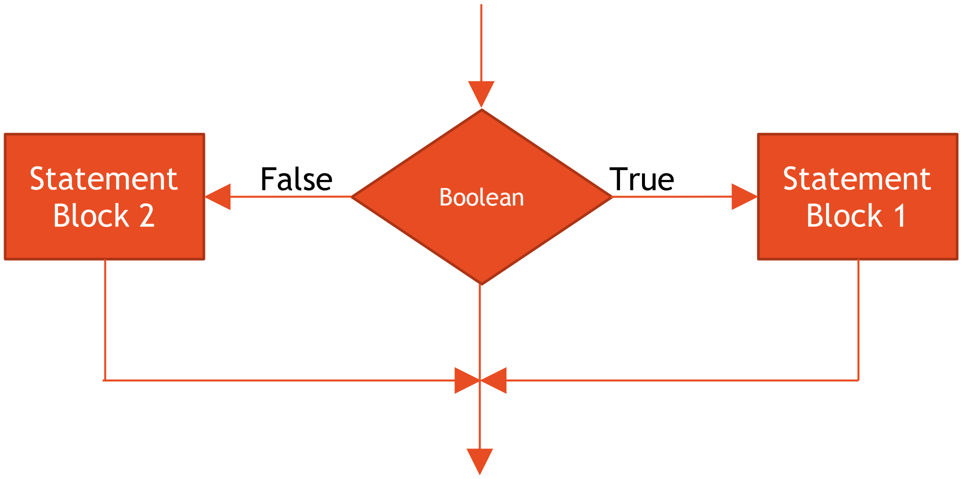
```
if condition:
    statement Block
```

02-05-if-absolute-value.py

```
def main():
    num = int(input('Enter an integer: '))
    if num < 0:
        num = -num

    print(f'The absolute value of your number is {num}')
```

if-else statement



```
if condition:
    Statement Block 1
else:
    Statement Block 2
```

02-06-if-else-count-odd-even.py

```python
def main():
    oddCount = evenCount = 0
    for i in range(100):
        x = random.randint(1,100)
        if x % 2 == 0:
            evenCount += 1
        else:
            oddCount +=1

    print(f'Odds: {oddCount}  Evens: {evenCount}')
```