

# Python Booleans and Logic



# Everyday True/False Questions

We use **True** or **False** all the time:

- Is it raining right now?  
→ True or False?
- Are you hungry?  
→ True or False?
- Do you have homework today?  
→ True or False?

Python can also store these **yes / no** answers using **Boolean**.

# What is a Boolean?

A **Boolean** is a value that is either `True` or `False`.

- It starts with a **capital** letter.
- No quotation marks.

# What is a Boolean?

A **Boolean** is a value that is either `True` or `False`.

- It starts with a **capital** letter.
- No quotation marks.

Example:

```
b = True          # b is the variable name, True is the value
print(b)
print(type(b))
```

# What is a Boolean?

A **Boolean** is a value that is either `True` or `False`.

- It starts with a **capital** letter.
- No quotation marks.

Example:

```
b = True          # b is the variable name, True is the value
print(b)
print(type(b))
```

Output:

```
True
<class 'bool'>
```

# Comparisons Make Booleans

These operators return `True` or `False`:

- `>` (greater than)
- `<` (less than)
- `>=` (greater than or equal to)
- `<=` (less than or equal to)
- `==` (equal to, double equals)
- `!=` (not equal)

Remember: `==` is **comparison**, `=` is **assignment**.

# Q1 True or False?

What is the output?

```
print(7 > 10)    # greater than  
print(3 == 3)   # equal to  
print(5 != 5)   # not equal to
```

# Q1 True or False?

What is the output?

```
print(7 > 10)    # greater than  
print(3 == 3)   # equal to  
print(5 != 5)   # not equal to
```

Output:

```
False  
True  
False
```



# Logical Operators: AND

**and** means both conditions must be **True**.

Example:

```
older_than_16 = True
passed_test = False

if older_than_16 and passed_test:
    print("You will get a driver license.")
else:
    print("You will not get a driver license.")
```

# Logical Operators: AND

**and** means both conditions must be **True**.

Example:

```
older_than_16 = True
passed_test = False

if older_than_16 and passed_test:
    print("You will get a driver license.")
else:
    print("You will not get a driver license.")
```

Output:

```
You will not get a driver license.
```

# Logical Operators: OR

**or** means **at least one** condition is **True**.

Example:

```
is_weekend = True
is_holiday = False

if is_weekend or is_holiday:
    print("You can stay at home.")
else:
    print("You will go to school.")
```

# Logical Operators: OR

**or** means **at least one** condition is **True**.

Example:

```
is_weekend = True
is_holiday = False

if is_weekend or is_holiday:
    print("You can stay at home.")
else:
    print("You will go to school.")
```

Output:

```
You can stay at home.
```

# Logical Operator: NOT

`not` means **opposite** ( `True` becomes `False`, `False` becomes `True` ).

Example:

```
raining = False  
  
if not raining:  
    print("Let's go to the park!")
```

# Logical Operator: NOT

`not` means **opposite** ( `True` becomes `False`, `False` becomes `True` ).

Example:

```
raining = False  
  
if not raining:  
    print("Let's go to the park!")
```

Output:

```
Let's go to the park!
```

Why? raining is `False` → not raining is `True` → print the message

## Q2 Movie Choice

What is the output?

```
age = 12
has_ticket = True

if age >= 13 and has_ticket:
    print("Teen movie")
else:
    print("Kids movie")
```

Is `age >= 13` True or False here?

## Q2 Movie Choice

What is the output?

```
age = 12
has_ticket = True

if age >= 13 and has_ticket:
    print("Teen movie")
else:
    print("Kids movie")
```

Output:

```
Kids movie
```

Why? `False and True` → `False` → go to `else`



## Q3 📦 Fill in the Blanks

You get a prize if **at least one** condition is True:

- `score` is at least 90,
- `has_coupon` is True.

```
score = 85
has_coupon = True

if _____:
    print("You get a prize!")
else:
    print("No prize this time.")
```

Output:

## Q3 📦 Fill in the Blanks

You get a prize if **at least one** condition is True:

- `score` is at least 90,
- `has_coupon` is True.

```
score = 85
has_coupon = True

if score >= 90 or has_coupon:
    print("You get a prize!")
else:
    print("No prize this time.")
```

Output:

```
You get a prize!
```

# Summary

- Boolean variables are either `True` or `False`.
- Comparisons ( `>`, `<`, `>=`, `<=`, `==`, `!=` ) return Booleans.
- `and` → both conditions must be True.
- `or` → at least one condition is True.
- `not` → flips True and False.
- Booleans are typically used in `if` and `elif` statements.