

WEEK 3 HOUR 1

TODAY'S TERMINOLOGY

Summary

- Boolean
- Comparison statements
- Boolean functions
- Conditional statements

TYPES OF VALUES

BOOL: BOOLEANS



Definition

A True or False value.



Warning

Python reserves the keyword **True** and **False**

BOOLEANS AND VARIABLES

STORING BOOLEANS



makeameme.org

COMPARISON OPERATORS

Operator	Symbol	Example Expression	Example Result
Less Than	<	$3 < 4$	True
Greater Than	>	$3 > 8$	False
Equal To	==	$7 \text{ == } 7$	True
Greater Than or Equal To	>=	$3.5 \text{ >= } 3.4$	True
Less Than or Equal To	<=	$3.5 \text{ <= } 3.4$	False
Not Equal To	!=	$4 \text{ != } 4$	False

LOGICAL OPERATORS

Symbol	Example Expression	Example Result
not	not (80 >= 50)	False
and	(grade >= 50) and (<u>grade2</u> >= 50)	True when both grade and <u>grade2</u> are 50 or more.
or	(grade >= 50) or (<u>grade2</u> >= 50)	True when at least one of grade and <u>grade2</u> are 50 or more.

OPERATORS PRECEDENCE

Operator	Description
<code>(expressions...), [expressions...], {key: value...}, {expressions...}</code>	Binding or parenthesized expression, list display, dictionary display, set display
<code>x[index], x[index:index], x(arguments...), x.attribute</code>	Subscription, slicing, call, attribute reference
<code>**</code>	Exponentiation [5]
<code>+x, -x, ~x</code>	Positive, negative, bitwise NOT
<code>*, @, /, //, %</code>	Multiplication, matrix multiplication, division, floor division, remainder [6]
<code>+, -</code>	Addition and subtraction
<code>in, not in, is, is not, <, <=, >, >=, !=, ==</code>	Comparisons, including membership tests and identity tests
<code>not x</code>	Boolean NOT
<code>and</code>	Boolean AND
<code>or</code>	Boolean OR
<code>if – else</code>	Conditional expression





Demo

- Numeric comparison
- Compare different types (e.g. int vs. float)
- Use variables to compare
- Logical and/or
- Logical not (precedence over and/or)

DEMO (LAZY EVALUATION)



makeameme.org

CONDITIONAL STATEMENTS

IF STATEMENT

ⓘ

Definition

Statement to evaluate whether an expression is true or not. **Only** if True, the code insight will run.



Example

```
is_open = True

if(is_open):
    # statement (will run)

# continue the code outside of if statement
```

Copy



CONDITIONAL STATEMENTS

ELSE STATEMENT



Definition

It follows an if statement. **Only** if condition in if statement is **False**, the code insight will run.



Example

```
is_open = False

if(is_open):
    # statement (will not run)
else:
    # other statement (will run)

# continue the code outside of if-else statements
```

Copy



CONDITIONAL STATEMENTS

ELIF STATEMENT



Definition

When we want to check for second (or more) conditions. **Only** if previous conditions are False and the current one is True, the code insight will run.



Example

```
is_open = False
is_free = False
x = 5
if(is_open):
    # statement 1 (will not run)
elif(is_free):
    # statement 2 (will not run)
elif(x < 10):
    # statement 3 (will run)
else:
    # other statement
```

Copy



IF, IF IF, IF-ELSE, IF-ELIF, IF-ELIF-ELSE,



WORKSHEET

IF STATEMENTS

