Chapter 7

Dr. Raza Ul Mustafa Khokhar
American University
Computer Science Department - CSC-148

Boolean Values and Boolean Expressions

```
print(True)
print(type(True))
print(type(False))
```

The Python type for storing true and false values is called bool, named after the British mathematician, George Boole. George Boole created *Boolean Algebra*, which is the basis of all modern computer arithmetic.

Boolean

-

```
print(5 == 5)

print(5 == 6)

j = "hel"
print(j + "lo" == "hello")
```

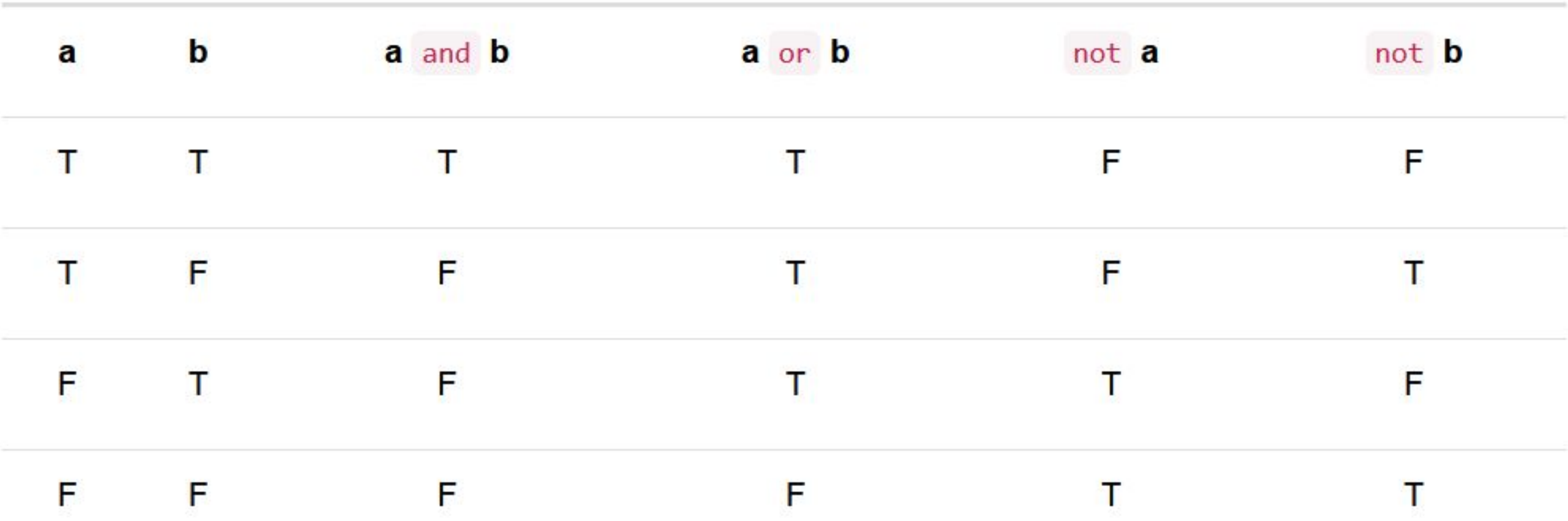
Six common comparison operators

```
x != y               # x is not equal to y
x > y                # x is greater than y
x < y                # x is less than y
x >= y               # x is greater than or equal to y
x <= y               # x is less than or equal to y
```

Logical operators

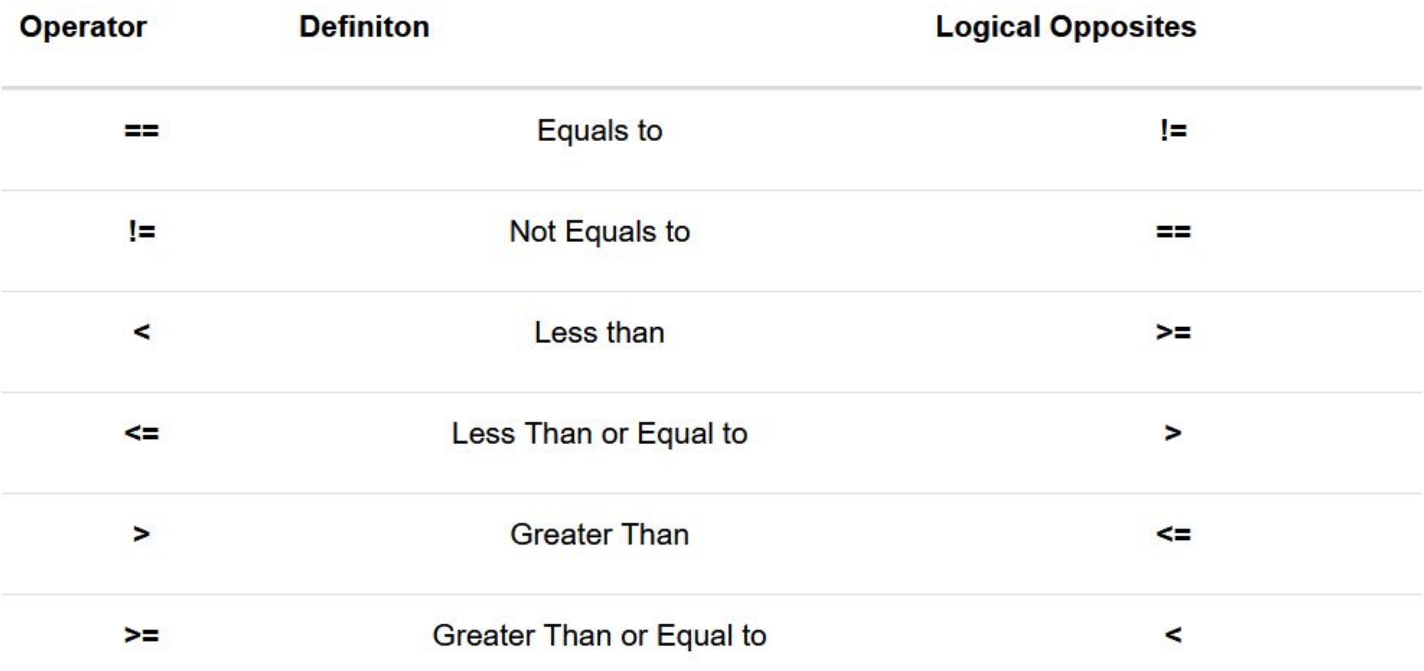
```
x = 5
print(x > 0 and x < 10)

n = 25
print(n % 2 == 0 or n % 3 == 0)
```

Logical operators



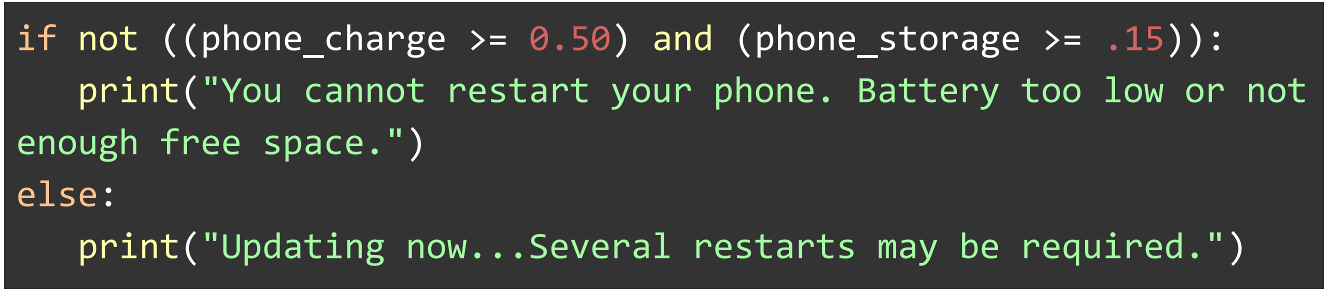
a	b	a `and` b	a `or` b	`not` a	`not` b
T	T	T	T	F	F
T	F	F	T	F	T
F	T	F	T	T	F
F	F	F	F	T	T

Logical opposites



Operator	Definiton	Logical Opposites
==	Equals to	!=
!=	Not Equals to	==
<	Less than	>=
<=	Less Than or Equal to	>
>	Greater Than	<=
>=	Greater Than or Equal to	<

Let’s look at some examples – AND



```python
if not ((phone_charge >= 0.50) and (phone_storage >= .15)):
    print("You cannot restart your phone. Battery too low or not enough free space.")
else:
    print("Updating now...Several restarts may be required.")
```

OR

```python
if (phone_charge < 0.50) or (phone_storage < .15):
    print("You cannot restart your phone. Battery too low or not enough free space.")
else:
    print("Updating now...Several restarts may be required.")
```

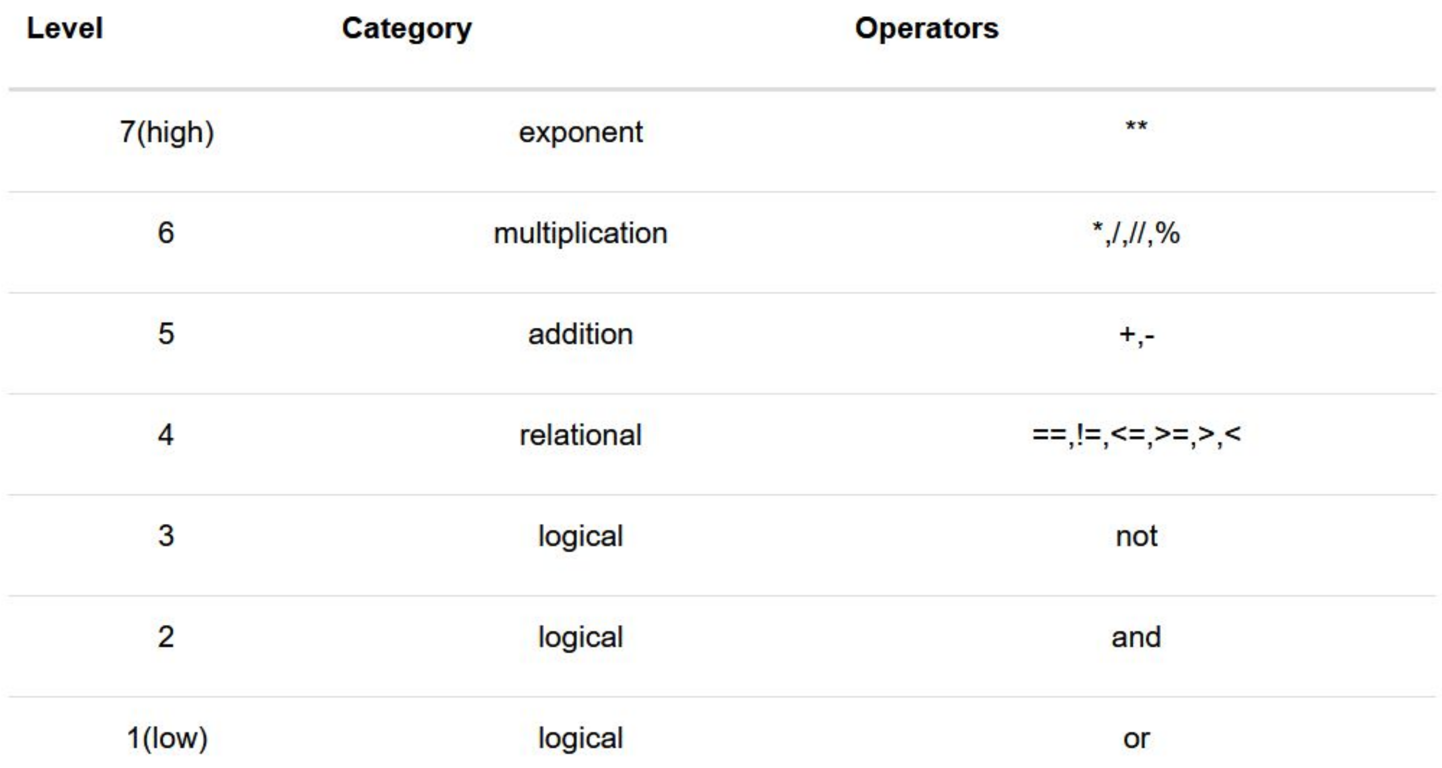
Precedence of Operators

- Python will always evaluate the arithmetic operators first (** is highest, then multiplication/division, then addition/subtraction)
- Next comes the relational operators. Finally, the logical operators are done last

```
6*5 >= 10 and 8-6 <= 20
```

- first perform the arithmetic and then check the relationships
- The and will be done last

Operator precedence table



Level	Category	Operators
7(high)	exponent	**
6	multiplication	*,/,//,%
5	addition	+,-
4	relational	==,!=,<=,>=,>,<
3	logical	not
2	logical	and
1(low)	logical	or

Conditional Execution: Binary Selection

Selection statements, sometimes also referred to as **conditional statements**, give us this ability. The simplest form of selection is the **if statement.** This is sometimes referred to as **binary selection** since there are two possible paths of execution

```
x = 15

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

Nested conditionals

One conditional can also be **nested** within another. For example, assume we have two integer variables, x and y. The following pattern of selection shows how we might decide how they are related to each other.

```
x = 10
y = 10

if x < y:
    print("x is less than y")
else:
    if x > y:
        print("x is greater than y")
    else:
        print("x and y must be equal")
```

Chained conditionals

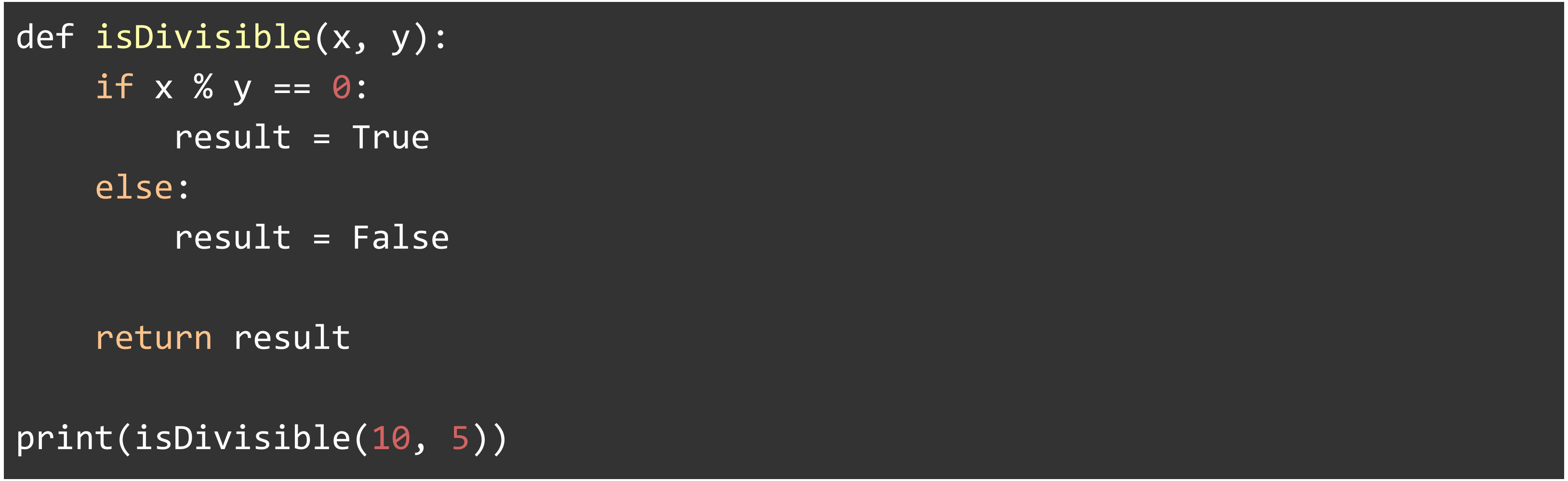
```python
x = 10
y = 10

if x < y:
    print("x is less than y")
elif x > y:
    print("x is greater than y")
else:
    print("x and y must be equal")
```

Nested-if, elif else

```python
score = 85

if score >= 90:
    print("A")
elif score >= 80:
    print("B")
elif score >= 70:
    print("C")
elif score >= 60:
    print("D")
else:
    print("F")
```

Boolean Functions

- We have already seen that boolean values result from the evaluation of boolean expressions.
- Since the result of any expression evaluation can be returned by a function (using the return statement), functions can return boolean values.



```python
def isDivisible(x, y):
    if x % y == 0:
        result = True
    else:
        result = False

    return result

print(isDivisible(10, 5))
```

Slides & Material

razaulmustafa.us/cs148/