

# CSCI 127: Joy and Beauty of Data

## Lecture 5: Selection & If Statements

Reese Pearsall

Snowmester 2020

<https://reese.github.io/classes/127/main.html>

# Announcements

Lab 2 due TONIGHT @ 11:59 PM

Program 1 posted (Due 12/9) @ 11:59 PM

-> Will need to understand iteration (Monday) to complete it

Lab 3 due on Monday @ 11:59 PM

-> will try to have it posted tonight or Saturday morning

## Today

Booleans, If statements



# Booleans

**Boolean** is a data type for storing true and false and values

There are only 2 values: `True` and `False`

# Boolean Comparison

**x == y**

# x is equal to y

4 == 4  
-> True

**x != y**

# x is not equal to y

3 != 4  
-> True

**x > y**

# x is greater than y

10 > 3  
-> True

**x < y**

# x is less than y

4 < 1  
-> False

**x >= y**

# x is greater than or equal to y

2 >= 3  
-> False

**x <= y**

# x is less than or equal to y

3 <= 3  
-> True

# Logical Operators

## and

Both conditions **must** be True

```
x = 5  
y = 4
```

```
print(x==5 and y==0)
```

False

```
x = 5  
y = 4
```

```
print(x==5 and y==4)
```

True

# Logical Operators

## or

One of the conditions must be True  
(or Both of the conditions must be True)

```
x = 5  
y = 4
```

```
print(x==5 or y==0)
```

True

```
x = 5  
y = 4
```

```
print(x==0 or y==1)
```

False

```
x = 5  
y = 4
```

```
print(x==5 or y==4)
```

True

# Practice Problems



## Checking a legal move in UNO

Consider a simplified game of UNO that has forty different cards. Each card has a color ("red", "yellow", "green" or "blue") and a value (0, 1, 2, 3, 4, 5, 6, 7, 8 or 9). Complete the boolean function below so that it returns True if the second card can be played on the first card and False otherwise.

```
def legal_play(first_value, first_color, second_value, second_color):
```

For example, `legal_play(3, "blue", 3, "green")` and `legal_play(5, "yellow", 7, "yellow")` should both return True, but `legal_play(9, "red", 6, "green")` should return False.

# Practice Problems

## Identifying leap years

The following criteria is used to determine whether a year is a leap year:

The year is evenly divisible by 4;

If the year can be evenly divided by 100, it is NOT a leap year, unless;

The year is also evenly divisible by 400. Then it is a leap year.

Write a function that takes a year as a parameter and returns True if the year is a leap year, False otherwise.