

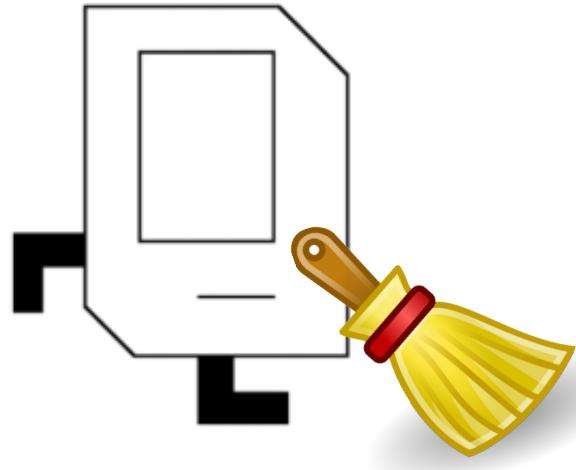


# Nested Structures

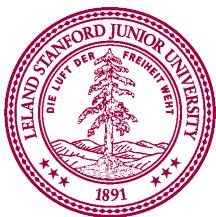
Chris Gregg

Based on Slides by Mehran Sahami and Chris Piech  
CS106A, Stanford University

# Housekeeping



- Assignment 5 goes out today!



# Why is this so fast?



mantis shrimp colors



All

Videos

Shopping

Images

News

More

Settings

Tools

About 1,870,000 results (0.54 seconds)

Humans and many other primates have three; some birds and reptiles have four photoreceptors. Certain butterflies can even have six. But the mantis shrimp has **12** different types of photoreceptors in their eyes – and scientists haven't understood why until now. Jan 27, 2014



[Study Offers Insights into Unique Color Vision of Mantis Shrimp ...](http://www.sci-news.com/biology/science-color-vision-mantis-shrimp-01719.html)  
[www.sci-news.com/biology/science-color-vision-mantis-shrimp-01719.html](http://www.sci-news.com/biology/science-color-vision-mantis-shrimp-01719.html)



# Review

# Core Datastructures

The standard is called “JSON”



All datasets can  
be represented by:

**Dictionaries,**  
**Lists,**  
strings,  
Floats,  
Integers  
Booleans,  
None,  
Blob



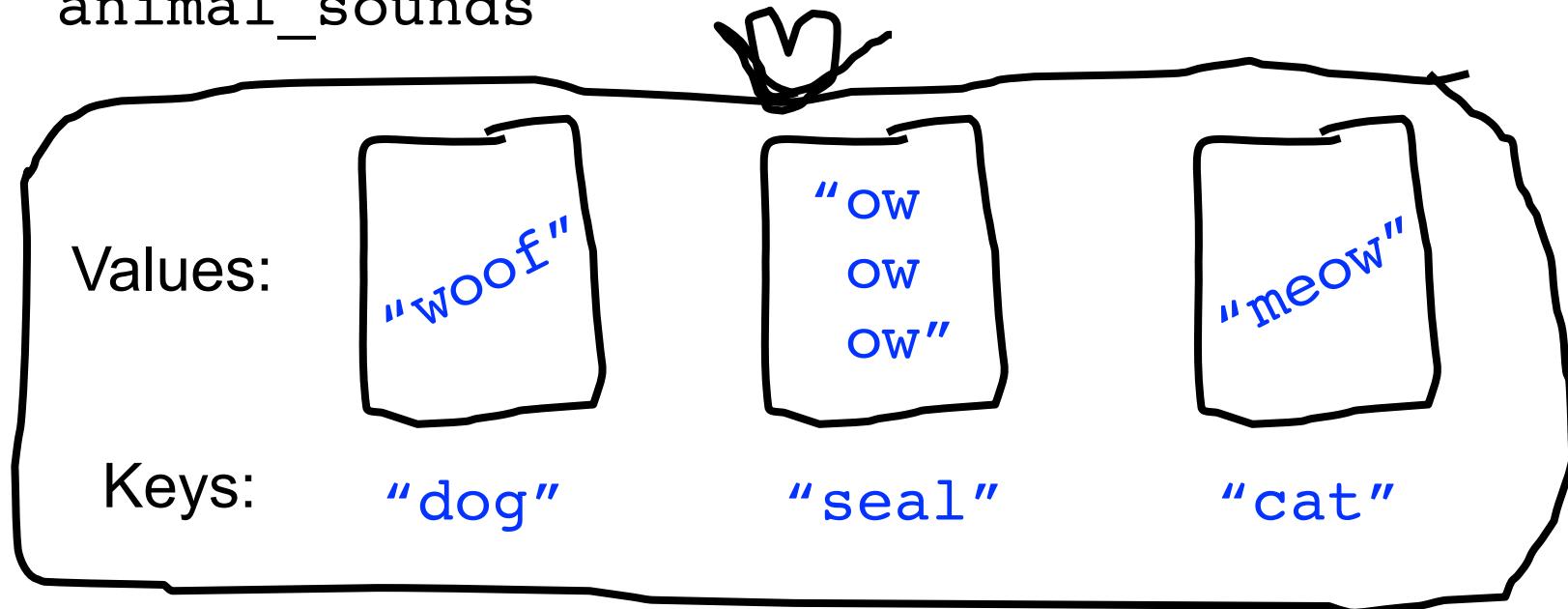
# Example Google Dicts Query Result

```
{  
  "markers": [  
    {  
      "name": "Rixos The Palm Dubai",  
      "position": [25.1212, 55.1535],  
    },  
    {  
      "name": "Shangri-La Hotel",  
      "location": [25.2084, 55.2719]  
    },  
    {  
      "name": "Grand Hyatt",  
      "location": [25.2285, 55.3273]  
    }  
  ]  
}
```



# Dict Review

animal\_sounds



# 1. Make a new Dict  
animal\_sounds = {}

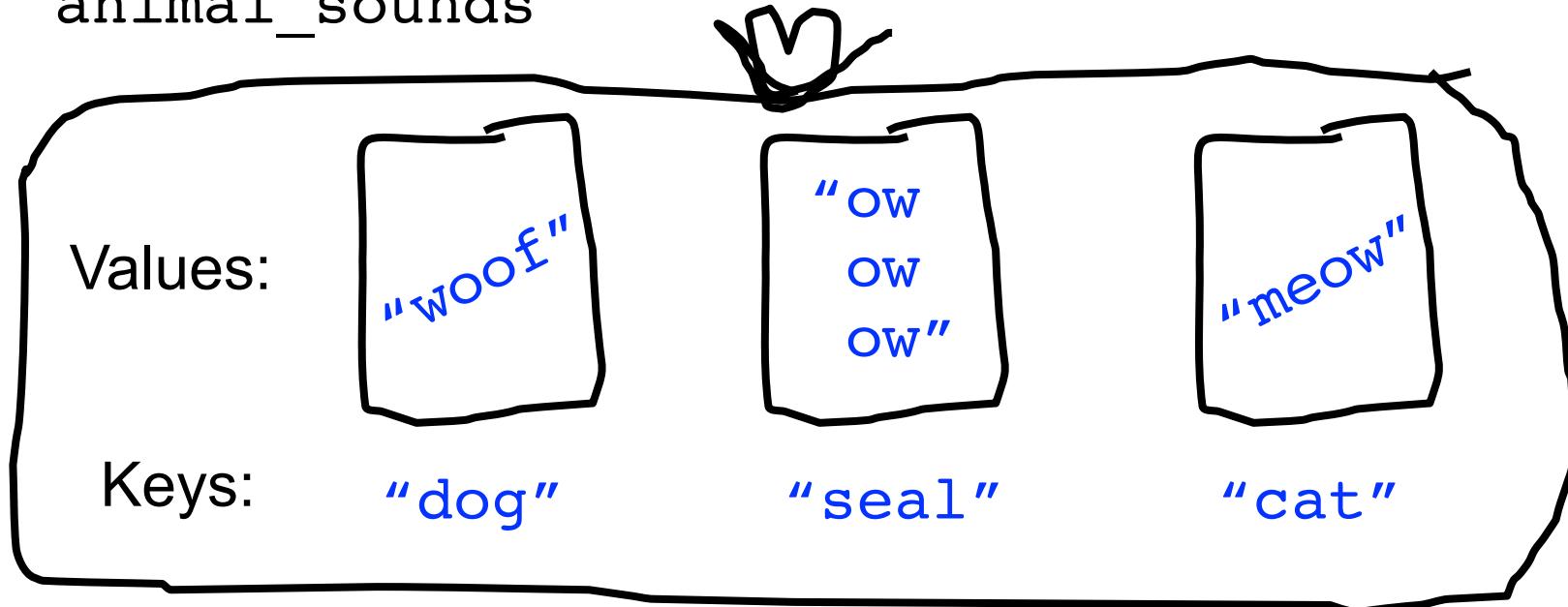
# 2. Put things into the Dict  
animal\_sounds["dog"] = "woof"  
animal\_sounds["cat"] = "meow"  
animal\_sounds["seal"] = "ow ow ow"

# 3. Get things out of the Dict  
dog\_sound = animal\_sound["dog"] # "woof"



# Dict Review

animal\_sounds



# 1. Make a new Dict  
animal\_sounds = {}

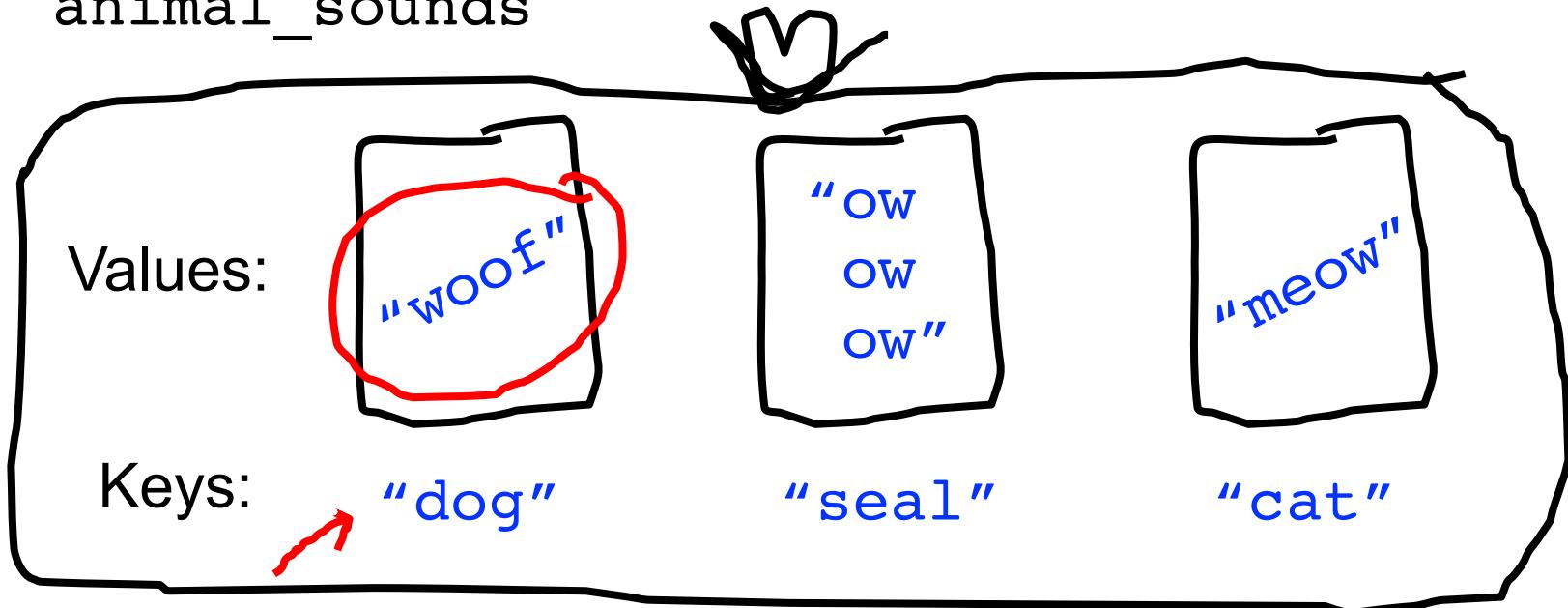
# 2. Put things into the Dict  
animal\_sounds["dog"] = "woof"  
animal\_sounds["cat"] = "meow"  
animal\_sounds["seal"] = "ow ow ow"

# 3. Get things out of the Dict  
dog\_sound = animal\_sound["dog"] # "woof"



# Dict Review

animal\_sounds



# 1. Make a new Dict  
animal\_sounds = {}

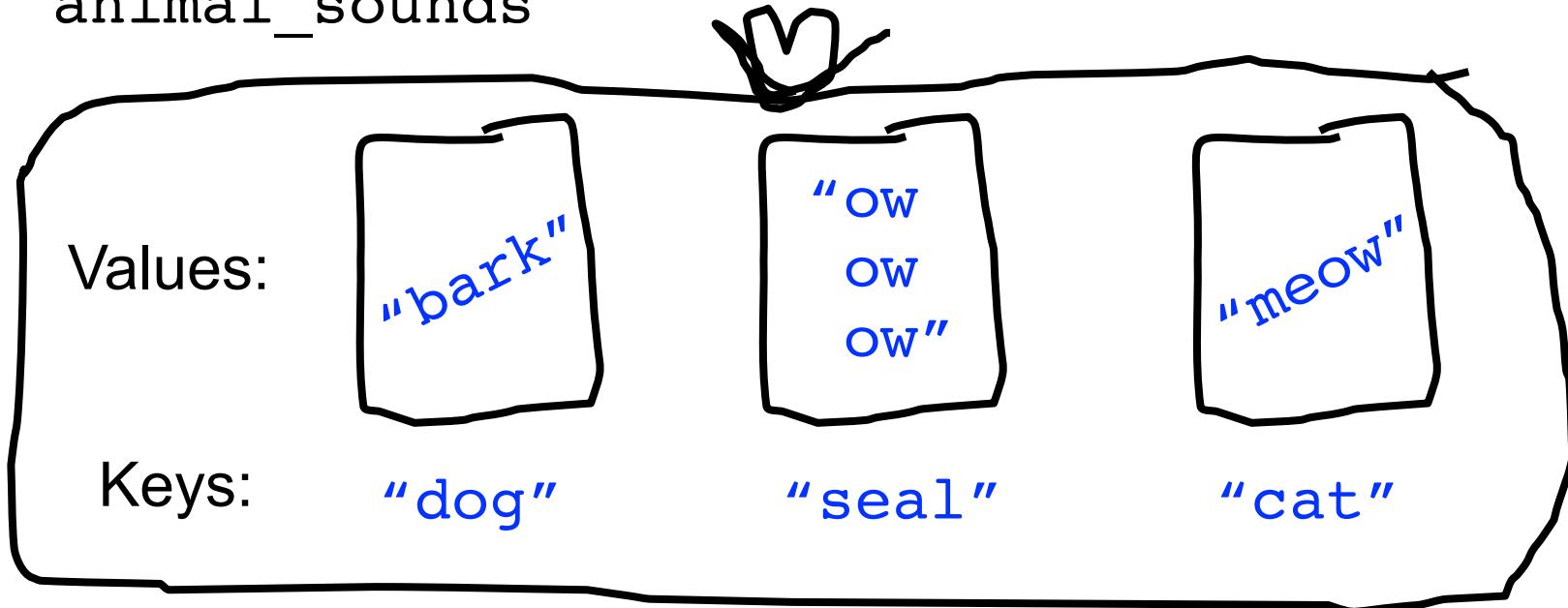
# 2. Put things into the Dict  
animal\_sounds["dog"] = "woof"  
animal\_sounds["cat"] = "meow"  
animal\_sounds["seal"] = "ow ow ow"

# 3. Get things out of the Dict  
dog\_sound = animal\_sound["dog"] # "woof"



# Dict Review

animal\_sounds



# 1. Make a new Dict  
animal\_sounds = {}

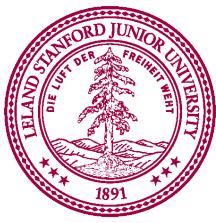
# 2. Put things into the Dict  
animal\_sounds["dog"] = "woof"  
animal\_sounds["cat"] = "meow"  
animal\_sounds["seal"] = "ow ow ow"

# 3. Get things out of the Dict  
dog\_sound = animal\_sound["dog"] # "woof"  
fox\_sound = animal\_sound["fox"] # Error: not in Dict



# Dictionary Recap

key → value



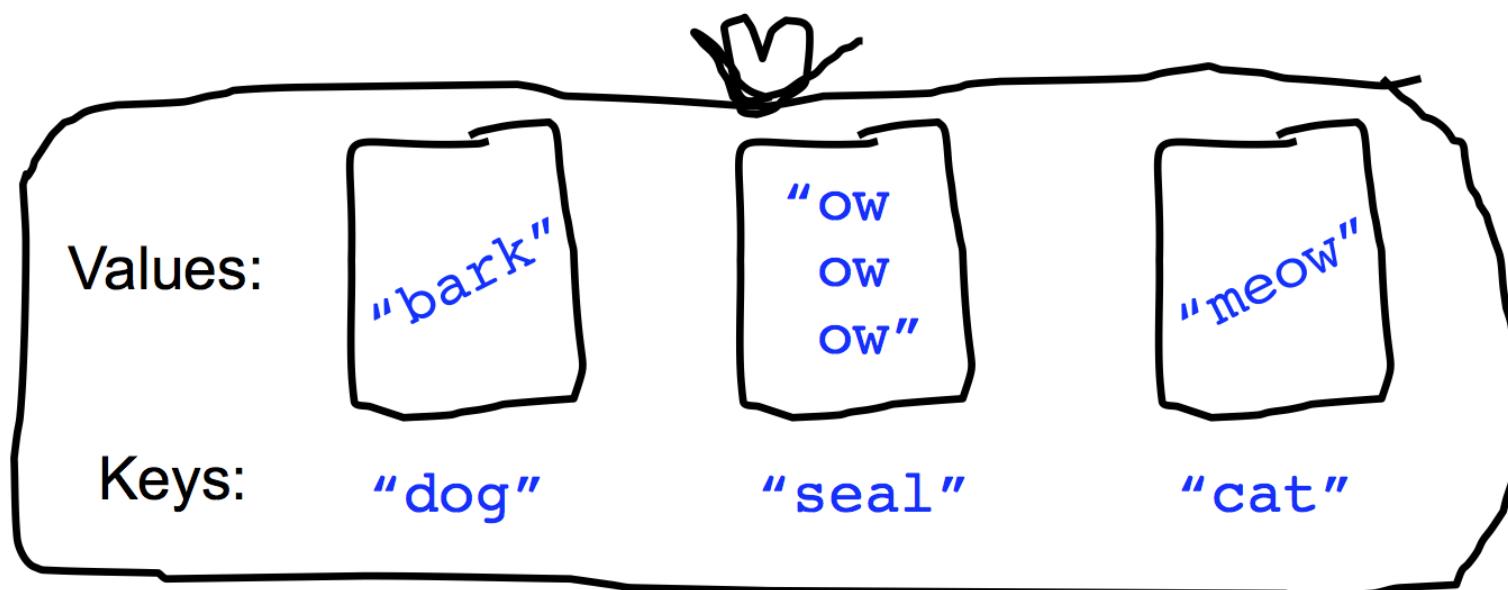
# Dictionary Recap

key



value

(string) animal → (string) animal sound

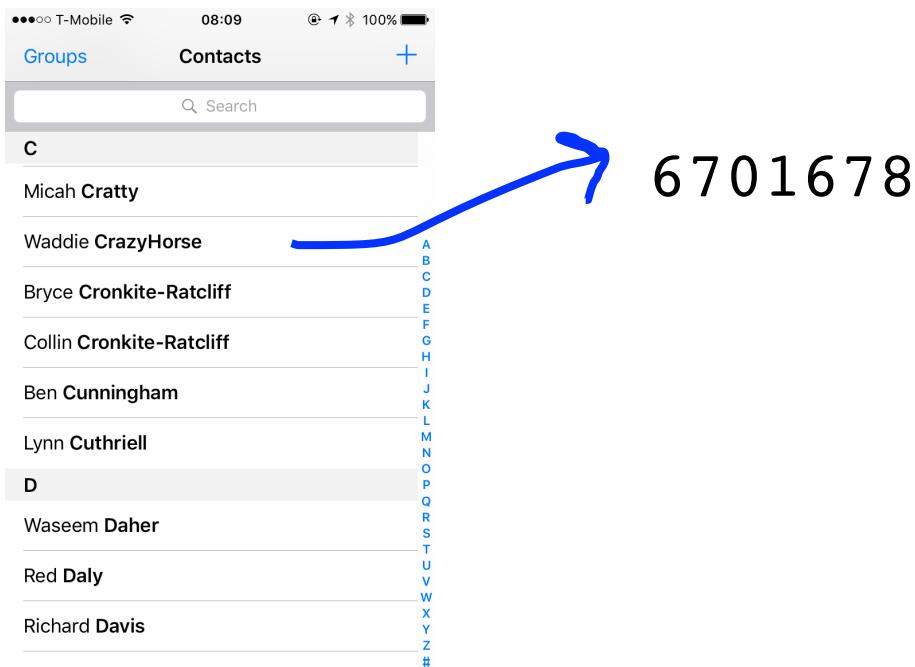


# Dictionary Recap

key → value

(string) animal → (string) animal sound

(string) name → (int) phone number



# Dictionarys on one slide

## 1. Make a Dictionary

```
my_Dict = {}
```

## 2. Put and get values into a Dict

```
my_Dict[key] = new_value  
my_Dict[key] # returns the corresponding value
```

## 3. Some useful other methods

```
size = len(my_Dict)  
key in my_Dict # returns true or false if key is in Dict
```

## 4. Iterate using a foreach loop

```
for key in my_Dict:  
    value = my_Dict[key] # look up the corresponding value
```





## Dictionaries are one way!

In dictionaries you can only look up values by keys.  
You can't look up keys by value.

animal\_sounds['meow'] → KeyError: 'b'





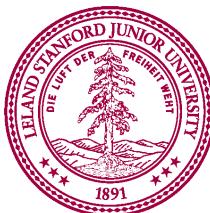
**Each key gets has only one value!**

If you put a key in the dictionary twice, it will overwrite

```
animal_sounds['dog'] = 'bark'  
animal_sounds['dog'] = 'woof'
```



```
animal_sounds = {'dog': 'woof'}
```



# Learned about Collections



List  
index -> value

Dictionary  
key -> value

# List

```
my_list = ['a', 'b', 'c']

print(my_list[1])

for i in range(len(my_list)):
    value = my_list[i]
    print(i, value)
```

my\_list

a	b	c
0	1	2

indices

# Dictionary

```
my_dict = {
    'x': 'a',
    'y': 'b',
    'z': 'c'
}

print(my_dict['y'])

for key in my_dict:
    value = my_dict[key]
    print(key, value)
```

my\_dict

a	b	c
'x'	'y'	'z'

keys



# List

```
my_list = [  
    'a',  
    'b',  
    'c'  
]  
  
print(my_list[1])  
  
for i in range(len(my_list)):  
    value = my_list[i]  
    print(i, value)
```

my\_list

a	b	c
0	1	2

indices

# Dictionary

```
my_dict = {  
    'x': 'a',  
    'y': 'b',  
    'z': 'c'  
}  
  
print(my_dict['y'])  
  
for key in my_dict:  
    value = my_dict[key]  
    print(key, value)
```

my\_dict

a	b	c
'x'	'y'	'z'

keys



# List

```
my_list = ['a', 'b', 'c']

print(my_list[1])

for i in range(len(my_list)):
    value = my_list[i]
    print(i, value)
```

my\_list

a	b	c
0	1	2

indices

# Dictionary

```
my_dict = {
    'x': 'a',
    'y': 'b',
    'z': 'c'
}

print(my_dict['y'])

for key in my_dict:
    value = my_dict[key]
    print(key, value)
```

my\_dict

a	b	c
'x'	'y'	'z'

keys



# List

```
my_list = ['a', 'b', 'c']

print(my_list[1])

for i in range(len(my_list)):
    value = my_list[i]
    print(i, value)
```

my\_list

a	b	c
0	1	2

indices

# Dictionary

```
my_dict = {
    'x': 'a',
    'y': 'b',
    'z': 'c'
}

print(my_dict['y'])

for key in my_dict:
    value = my_dict[key]
    print(key, value)
```

my\_dict

a	b	c
'x'	'y'	'z'

keys



# List

```
my_list = ['a', 'b', 'c']

print(my_list[1])

for i in range(len(my_list)):
    value = my_list[i]
    print(i, value)
```

my\_list

a	b	c
0	1	2

indices

# Dictionary

```
my_dict = {
    'x': 'a',
    'y': 'b',
    'z': 'c'
}

print(my_dict['y'])

for key in my_dict:
    value = my_dict[key]
    print(key, value)
```

my\_dict

a	b	c
'x'	'y'	'z'

keys



End Review

Are you ready?

For...

The ULTIMATE cs106a question?

# Ultimate CS106A: Reverse a Dict



**Normal Dict:**

Key -> Value



**Reversed Dict:**

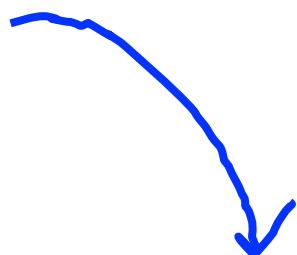
Value -> Keys

Claim: understanding this single example is most indicative of mastery in CS106A



# Ultimate CS106A: Reverse a Dict

```
ages = {  
    'Snoopy':50,  
    'Gary':70,  
    'Ada':32,  
    'Wil':23,  
    'Adele':32,  
    'Lionel':32,  
    'Rihanna':32,  
    'Stephen':32  
}
```



```
reversed = {  
    50:[['Snoopy'],  
        ['Gary'],  
        ['Ada', 'Adele', 'Lionel', 'Rihanna', 'Stephen'],  
        ['Wil']]  
}
```



To the code!!!

# Ultimate CS106A: Reverse a Dict

Gary -> 70

```
reversed = {
```

```
}
```



# Ultimate CS106A: Reverse a Dict

Gary -> 70

```
reversed = {
```

```
    70 : ['Gary'],
```

```
}
```



# Ultimate CS106A: Reverse a Dict

Ada -> 32

```
reversed = {
```

```
    70 : ['Gary'],
```

```
    32 : ['Ada'],
```

```
}
```



# Ultimate CS106A: Reverse a Dict

Snoopy -> 50

```
reversed = {
```

```
    70 : ['Gary'],
```

```
    32 : ['Ada'],
```

```
}
```



# Ultimate CS106A: Reverse a Dict

Snoopy -> 50

```
reversed = {  
    70 : ['Gary'],  
    50 : ['Snoopy'],  
    32 : ['Ada'],  
}
```



# Ultimate CS106A: Reverse a Dict

Rihanna -> 32

```
reversed = {  
    70 : ['Gary'],  
    50 : ['Snoopy'],  
    32 : ['Ada'],  
}
```



# Ultimate CS106A: Reverse a Dict

Rihanna -> 32

```
reversed = {  
    70 : ['Gary'],  
    50 : ['Snoopy'],  
    32 : ['Ada', 'Rihanna'],  
}
```



# Ultimate CS106A: Reverse a Dict

```
reversed = {
```

```
    70 : ['Gary'],
```

```
    50 : ['Snoopy'],
```

```
    32 : ['Ada', 'Rihanna', 'Stephen', ... ]
```

```
    23 : ['Wil']
```

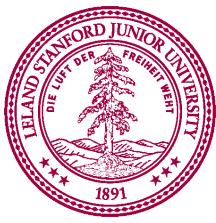
```
}
```



something awesome

\*idea credits to Keith

# The XKCD Color Survey









# The XKCD Color Survey

- Volunteers (online) were shown a randomly-chosen color and asked to name the color.
- The result is (after filtering) about 2.8 million RGB triplets and their names.
- What do people think the colors are?



# The File Format

## *color-name, red, green, blue*

navy blue,27,34,98  
blue,41,201,234  
lime green,99,212,32  
red brown,160,89,66  
orange,204,117,64  
teal,12,208,219  
blue,73,97,236  
dark tan,209,202,95  
moss green,77,147,83  
magenta,136,30,75  
blue,33,115,229  
**goldenrod**,232,171,51  
purplish blue,99,46,219  
gray,212,209,208  
green,56,188,125  
mustard,197,164,25  
red,242,9,26  
pale green,221,240,210  
cyan,199,254,247  
**carrot**,240,80,16  
purple,186,117,237  
pale rose,197,68,63

**fuchsia**,210,13,137  
pea green,198,247,15  
forest green,17,106,39  
tan,173,163,123  
dark blue,27,7,117  
teal,41,182,127  
aqua,36,219,173  
dark green,17,110,73  
pale lime,189,244,125  
light green,115,235,119  
bright blue,17,155,238  
hot pink,247,3,229  
**lighter green**,98,253,147  
brown,138,112,77  
purple,116,50,76  
red,245,42,54  
green,7,173,31  
bluish gray,82,110,127  
blue,124,164,176  
blue,120,158,209  
**sand**,235,175,100  
forest green,32,144,58

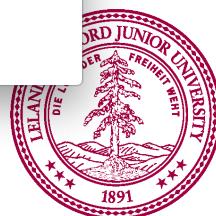
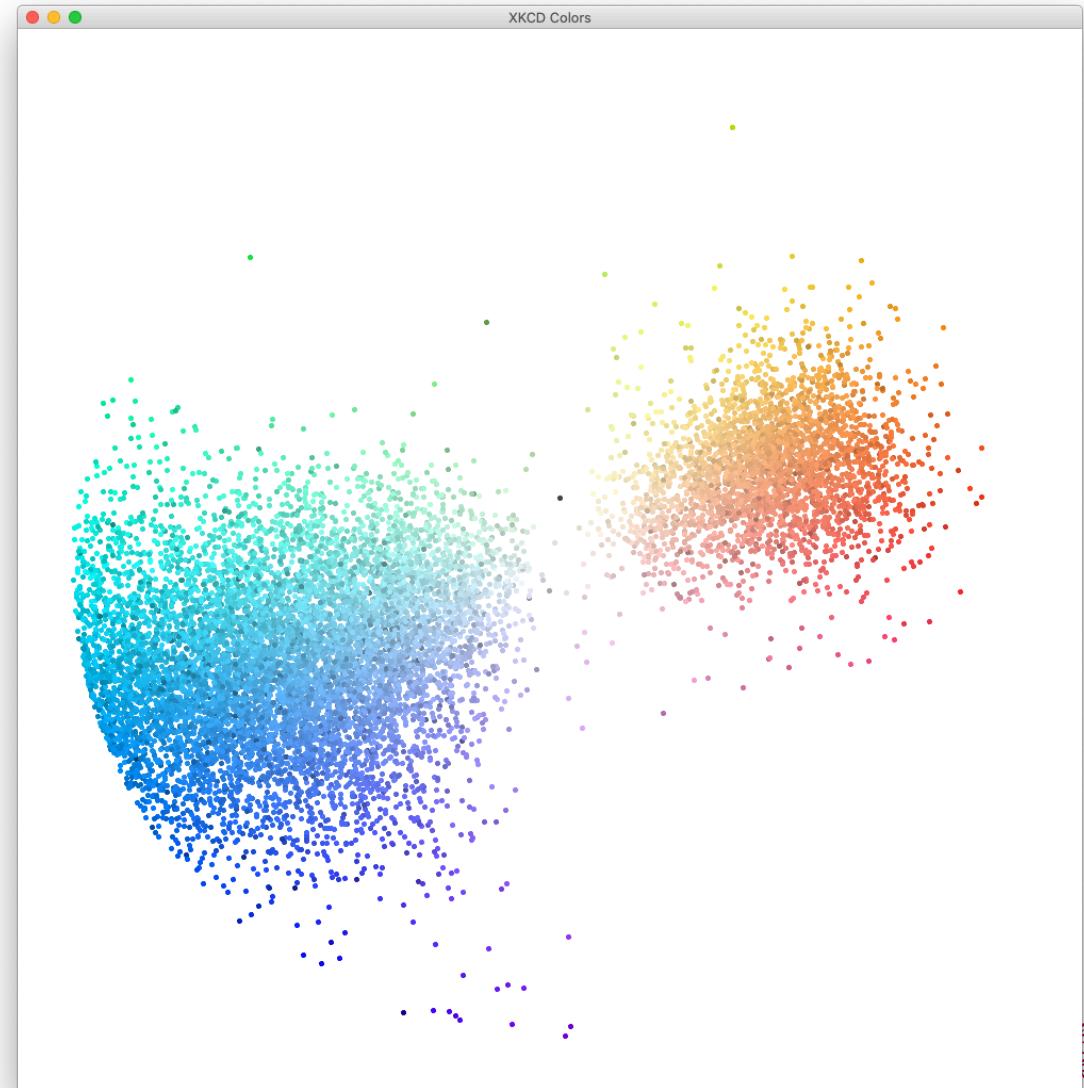
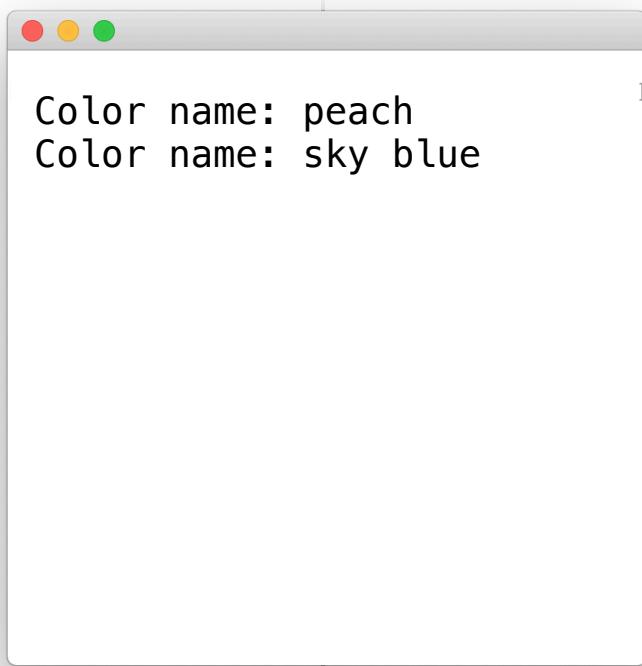
purple,145,37,226  
dirty green,87,130,64  
**dirty green**,125,136,42  
brown,132,116,30  
pink,252,68,255  
blueberry,71,55,114  
yellow brown,179,163,23  
purple,199,64,183  
deep purple,95,21,87  
**dirty yellow**,221,198,107  
light purple,185,110,194  
sea blue,24,250,209  
navy blue,16,32,75  
bluish green,62,208,104  
dark blue,2,0,50  
blue,107,148,220  
dark blue,101,68,175  
sky blue,7,152,170  
**teal**,81,166,152  
green,19,246,59  
green,20,252,59  
**aquamarine**,65,206,  
163



# How to Structure Data?

I give this to  
you so you can  
focus on data

```
def plot_color(canvas, r, g, b):
```



# How to Structure Data

blue	→	15 137 255	0 0 127	88 88 190
pink	→	166 14 7	99 55 5	255 0 0
gray	→	154 156 157	243 242 254	140 143 148

*associate each color name  
with a list of colors*



# How to Structure Data

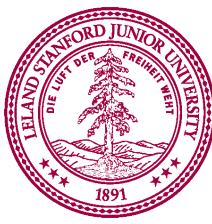
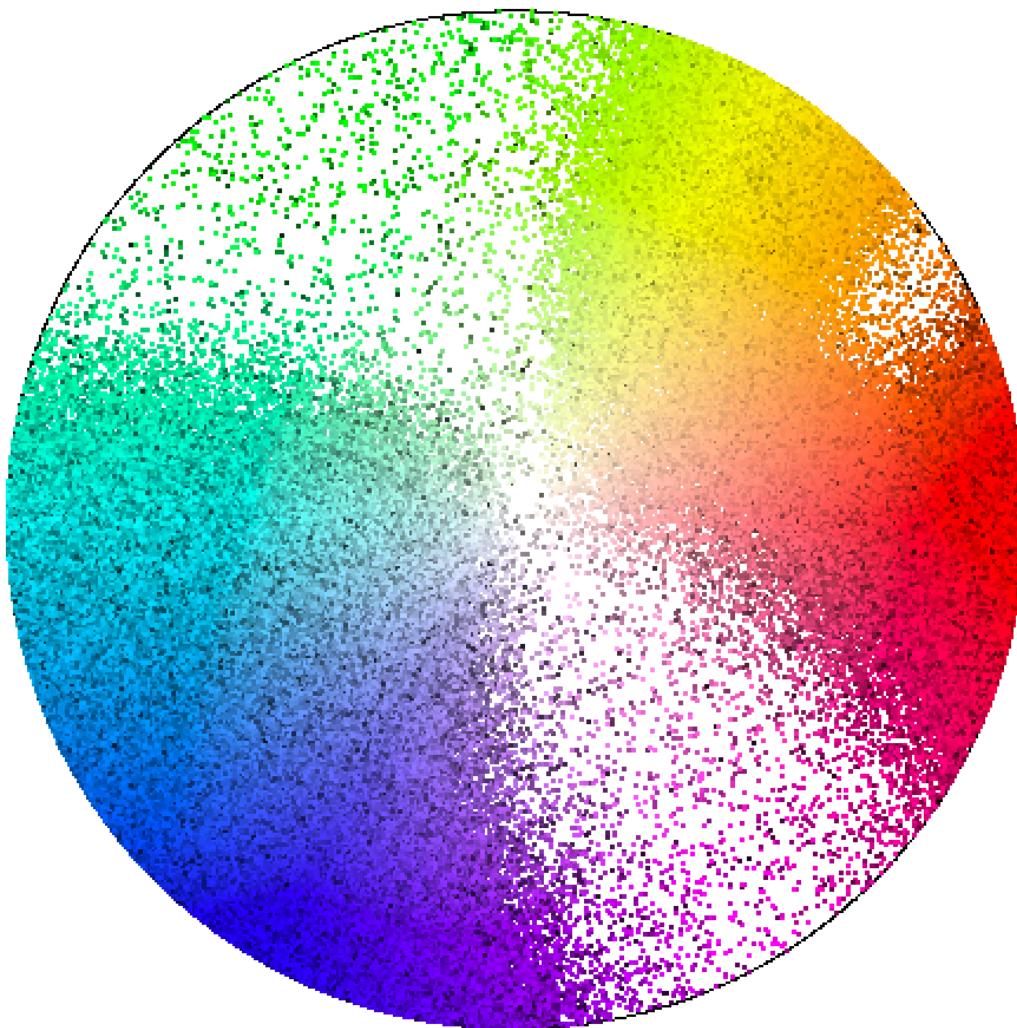
{

    "clover green": [[100, 216, 135], [72, 218, 111],  
        [57, 109, 40], [9, 190, 78], [4, 217, 90], [36,  
            164, 33], [85, 195, 120], [137, 207, 101], [155,  
            213, 167], [41, 141, 12], [35, 195, 118], [63,  
            169, 115], [2, 184, 86], [49, 189, 100], [147,  
            200, 8], [63, 160, 43], [87, 121, 8], [49, 183,  
            44], [61, 190, 119]],  
    "sal": [[184, 207, 244], [48, 199, 109], [247, 4,  
        25], [6, 101, 127], [196, 124, 36], [148, 30, 23],  
        [106, 51, 249], [186, 63, 96], [209, 234, 226],  
        [115, 18, 254], [59, 251, 10], [209, 84, 209],  
        [254, 164, 39], [154, 165, 137], [82, 196, 178],  
        [120, 250, 248], [175, 59, 33], [67, 52, 126],  
        [224, 211, 50], [9, 255, 249], [138, 43, 154],  
        [218, 158, 7], [213, 79, 90]],  
    "marzipan": [[202, 197, 102], [34, 80, 112], [127,  
        162, 51], [90, 171, 24], [134, 198, 156], [163,  
            138, 126], [212, 248, 154], [133, 25, 118], [75,  
            143, 86], [46, 108, 0], [9, 242, 107], [29, 120,  
            25], [237, 209, 155], [215, 82, 187], [200, 79,  
            52], [12, 78, 60], [18, 52, 183], [186, 61, 232],  
            [169, 201, 232], [173, 216, 142]]

}



# Displaying Colors



# Further Reading

- [http:#blog.xkcd.com/2010/05/03/color-survey-results/](http://blog.xkcd.com/2010/05/03/color-survey-results/)



# Why is this so fast?



mantis shrimp colors



All

Videos

Shopping

Images

News

More

Settings

Tools

About 1,870,000 results (0.54 seconds)

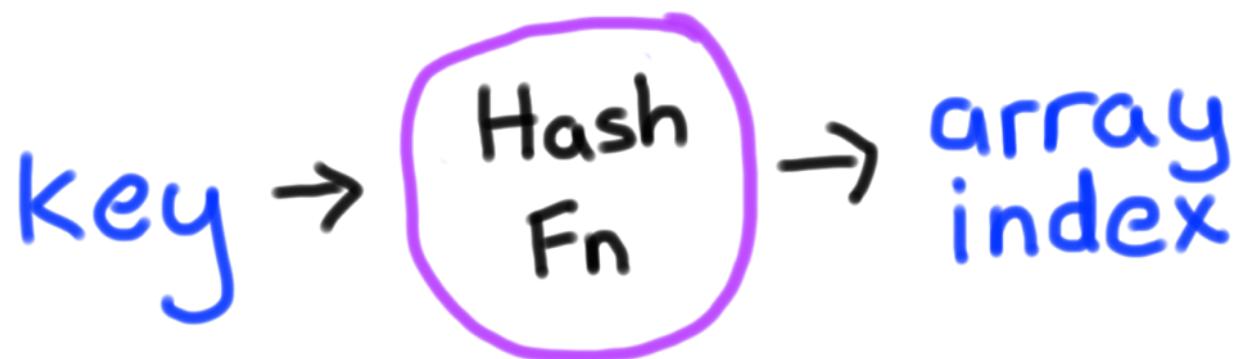
Humans and many other primates have three; some birds and reptiles have four photoreceptors. Certain butterflies can even have six. But the mantis shrimp has **12** different types of photoreceptors in their eyes – and scientists haven't understood why until now. Jan 27, 2014



[Study Offers Insights into Unique Color Vision of Mantis Shrimp ...](http://www.sci-news.com/biology/science-color-vision-mantis-shrimp-01719.html)  
[www.sci-news.com/biology/science-color-vision-mantis-shrimp-01719.html](http://www.sci-news.com/biology/science-color-vision-mantis-shrimp-01719.html)



# Why is this so fast?

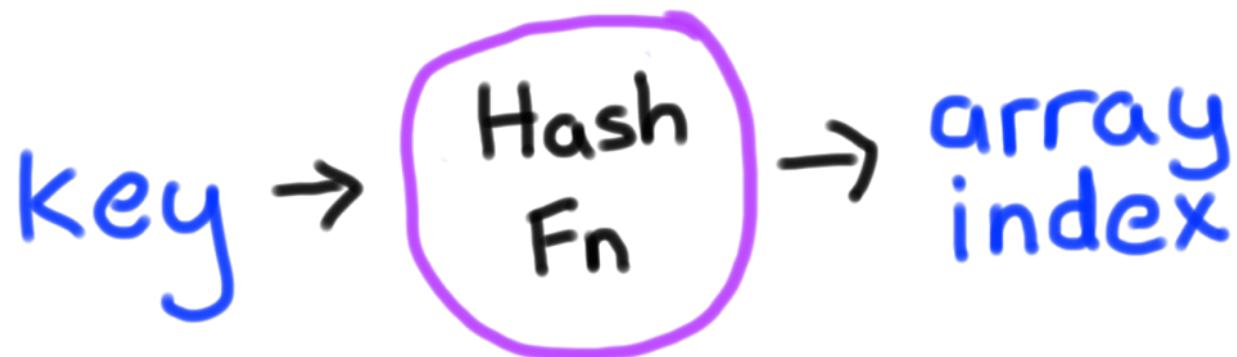


```
hash_int = hash(key);
```

\* Learn more in CS106B



# Why is this so fast?



```
hash_int = hash(key);
```

(but we lose sortedness)

\* Learn more in CS106B

