Python Bootcamp

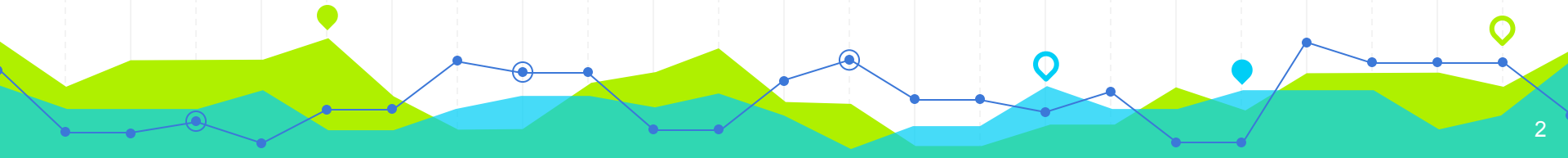
Workshop 4

8/11

Youth In Code
Justin Liu

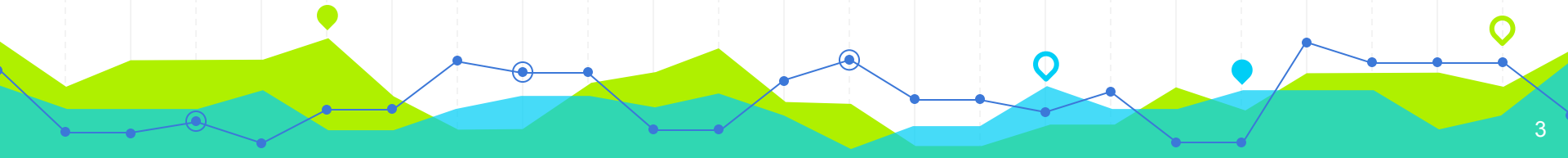
Roadmap

- Dictionaries
- Dictionary Functions
- Review
- 40 min. cumulative asynchronous lab on lists and dictionaries



Dictionaries

- A mutable, unordered data structure specific to Python
- Operates with keys and values instead of indexes
- Each value (of any data type) stored in a dictionary can be accessed using a key
- Cannot be sliced
- Indicated by {}
- keys(), values(), and items() don’t return true lists, but we can think of it as so



Dictionaries

```
>>> spam = {'color': 'red', 'age': 42}
>>> for v in spam.values():
        print(v)
Output:
red
42

>>> for k in spam.keys():
        print(k)
Output:
color
age
```

```
>>> for i in spam.items():
...     print(i)
Output:
('color', 'red')
('age', 42)
```

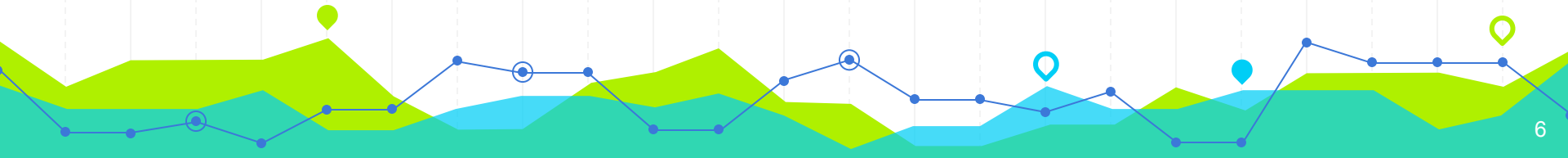
Dictionary Functions

- Although dict[‘key’] is what you will most likely be using most, there are some other functions worth noting
- get(key, fallback_value)
  - Checks if a key exists in a dictionary before accessing that key’s value
  - Sometimes better than dict[‘key’] if you’re not sure if that key exists
- setdefault(key_to_check, default_val) is a shortcut to ensure a key exists
- in and not in can be applied to dictionaries’ keys and values as well
- There is no add() or append() function for dictionaries
  - Instead, you simply make an assignment for the new data: dict[key] = value

Dictionaries (example)

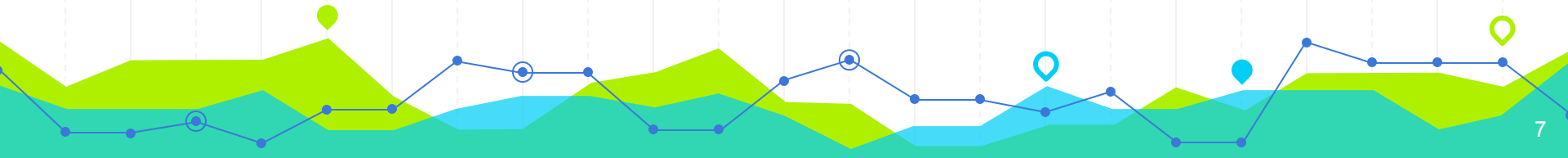
```
myCat = {'size': 'fat',
         'color': 'gray',
         'disposition': 'loud'}
```

- keys are 'size', 'color', and 'disposition'
- values are 'fat', 'gray', and 'loud'



Review

Within the span of only 4 workshops, you now know about variables and data types, taking in user input, flow control, the Random module, various loops, functions, lists, tuples, and dictionaries. Give yourself a pat on the back!



Practice Problems

1. Create a list of 10 randomly chosen numbers. Loop through the list and find the sum of only the even numbers in the list.
2. Create a dictionary of movies, where the key is the movie name and the value is your personal rating of the movie on a scale of 1 (atrocious) - 5 (would watch again). Iterate through all the movies in the dictionary, printing both its name and your rating in the format: “On a scale from 1 to 5, I rate the [insert movie_name] a [insert your_rating].”
3. Continuing on from 2), ask the user to input a movie name and their rating of it (1-5). If that movie name is already in the dictionary, print(‘Duplicate movie.’). Otherwise, add the movie name and its rating into the dictionary.
4. (Optional): Put all of this code into a loop such that once the user wants to stop inputting movies, you terminate the program. You can do this by prompting the user for a yes/no response regarding if they want to “continue” each time after they enter a new movie.